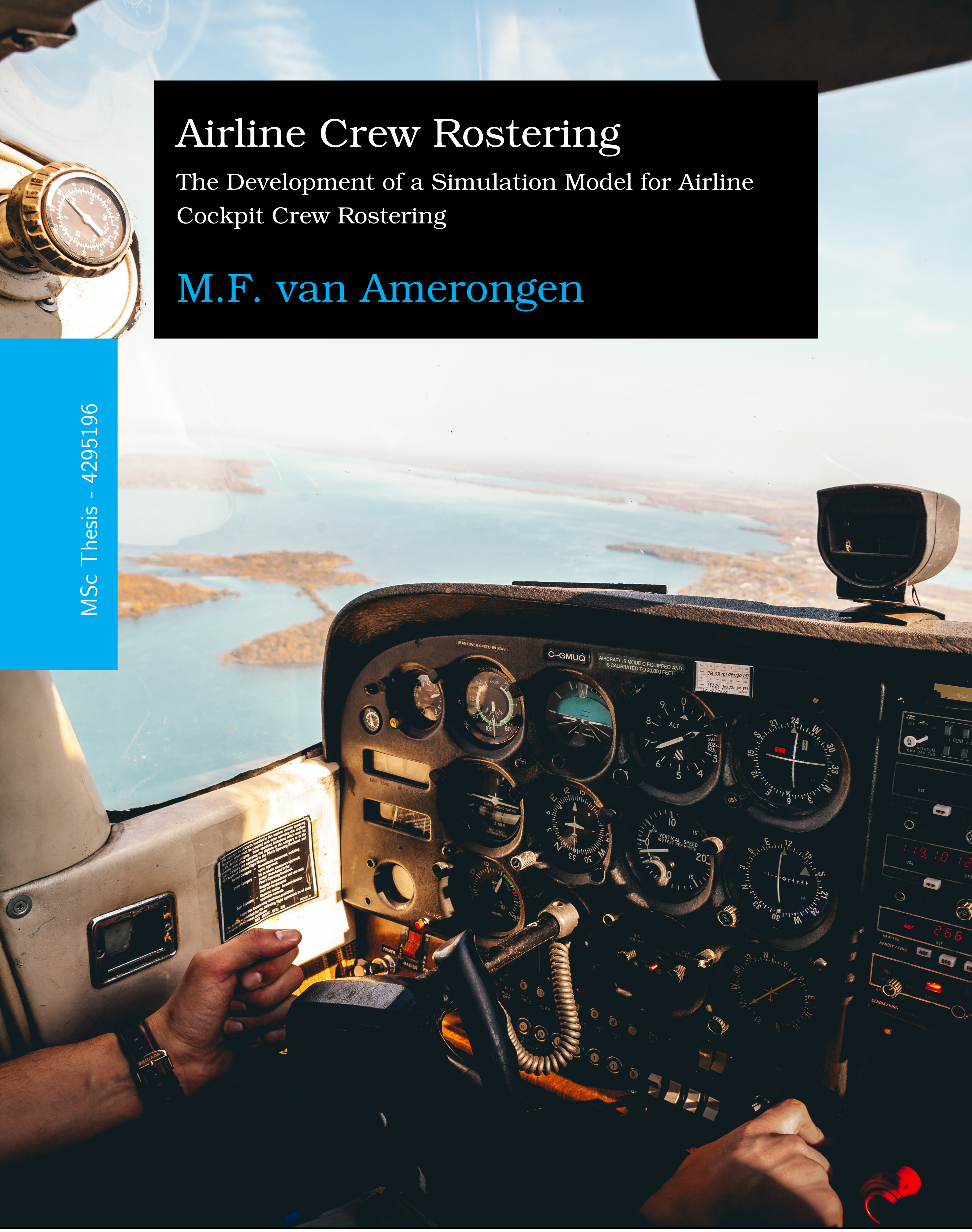

Airline Crew Rostering

The Development of a Simulation Model for Airline Cockpit Crew Rostering

M.F. van Amerongen

MSc Thesis - 4295196

Delft University of Technology

Aerospace Engineering

July 26, 2019

Delft University of Technology

Airline Crew Rostering

The Development of a Simulation Model for Airline Cockpit Crew Rostering

by

M.F. van Amerongen

To obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday August 9, 2019 at 13:00 PM.

Student number:	4295196	
Project duration:	October 1, 2018 – July 26, 2019	
Thesis committee:	Dr. ir. W.J.C. Verhagen,	TU Delft
	Dr. ir. B.F. Lopes dos Santos,	TU Delft
	Dr. ir. E. van Kampen,	TU Delft
	Ir. drs. L. Scherp,	KLM Royal Dutch Airlines

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Frontpage image credit: Photo by William Topa on Unsplash (https://unsplash.com/search/photos/cockpit)

Preface

This report is the final result of the research project that marks the end of my studies in Aerospace Engineering at the TU Delft. For the past ten months, I have had the opportunity to perform my research in collaboration with KLM Royal Dutch Airlines and to study one of the most complex mathematical optimization problems encountered in the airline industry.

I am very grateful for the opportunity to perform this challenging project and I would like to thank a number of people in particular for their continuous support throughout my thesis project.

First of all, I would like to thank my daily supervisor, Lennart Scherp, for his guidance, commitment and enjoyable company. Your care for the group of graduation students at the KLM Operations Control Center is greatly appreciated. Secondly, my professor and supervisor, Bruno Santos, for the interesting discussions and constructive feedback that were of great value in finding my research direction. Thirdly, I would like to thank my colleagues at KLM who have been involved in the project. A special thanks to Nico Scheeres, Marco van Vliet and Shirah van den Hoek for providing valuable insights on the practical challenges at KLM Cockpit Crew Services. Fourthly, I would like to thank the other TU Delft students at the KLM Operations Control Center, Maud Beulen, Toine Hooijen en Joey van Kempen, for their great company during my time at KLM. My final thanks goes to my family and friends, who have been incredibly supportive throughout my studies.

Michiel van Amerongen
Amsterdam, July 2019

Contents

Part I

Paper

Development of a Simulation Model for Airline Cockpit Crew Rostering

M.F. van Amerongen (MSc Student)
Supervisors: Ir. drs. L. Scherp, Dr. ir. B.F. Lopes dos Santos
Section Air Transport & Operations, Department Control and Operations, Faculty of Aerospace Engineering, Delft University of Technology, Delft, The Netherlands

***Abstract*— This paper presents the development of a simulation model for airline cockpit crew rostering. The developed simulation model supports airlines in performing comparative analyses of different crew rostering scenarios. The model was developed in collaboration with a major European airline, leveraging actual historical pairing, crew and roster data to accurately model the airline crew rostering process. The simulation model is based on a novel solution method that is based on the Hungarian assignment algorithm and that exploits the resemblance of the weekly crew rostering problem to a minimum-cost bipartite matching problem. An assessment of the performance of the simulation model has shown that the simulation model is able to construct rosters of near-optimal solution quality in limited computational time. To test the applicability of the developed model as a scenario analysis tool, two experiments have been performed. The insights the performed experiments have provided confirm the practical value of the developed simulation model.**

***Index Terms*— Airline Crew Rostering Problem, Personalized Crew Rostering, Simulation Model**

I. INTRODUCTION

Airline crew rostering is a sub-problem of the airline crew scheduling problem and involves the assignment of pairings, i.e. efficient sequences of flights, to individual crew members. The main objective is to make sure that every pairing is assigned to a crew member. Leaving pairings open induces high costs for airlines since these open pairings need to be resolved with expensive overtime payment, the hiring of freelancers or by cancelling the flight (Kohl & Karisch, 2004). Other objectives in the rostering process include constructing rosters with desirable working patterns, satisfying crew requests for activities or days-off and ensuring an equal workload among crew members (Kohl & Karisch, 2004). These objectives are pursued within the boundaries of rostering policies stemming from labour union agreements. With the increasing number of flights that need to be assigned and the growing number of labour union agreements to be respected, efficiently solving the crew rostering problem has become a major challenge for airlines (Eltoukhy, Chan, & Chung, 2017; Maenhout & Vanhoucke, 2010).

Academic literature on airline crew rostering has been focused on developing more efficient solution methods for the crew rostering problem. Over the past thirty years, important advancements have been made on developing more efficient heuristics and mathematical programming techniques to solve crew rostering problems of increasing size and complexity (Gamache, Soumis, & Marquis, 1999; Kasirzadeh, Saddoune, & Soumis, 2017; Ryan, 1992). However, besides actually solving the rostering problem, a major challenge faced by airlines is to determine the appropriate conditions in which the crew rostering problem is solved. In academic literature, the conditions in which the airline crew rostering problem is solved is considered a rigid framework, taken into account as fixed input into a crew rostering model. However, in reality, the crew composition, flight schedule and rostering policies of an airline evolve over time due to changes in airline policies and labour union agreements.

Changes in crew composition, flight schedule and labour union agreements have a major impact on the efficiency of the crew rostering process. Without a thorough understanding of the impact of these changing rostering conditions, airlines run the risk of agreeing to long-term commitments that significantly deteriorate the roster efficiency, resulting in increased personnel costs.

In order for airlines to make substantiated and data-driven decisions on crew composition and rostering policies, airlines require scenario analyses that quantify the impact of changing rostering conditions. However, due to the sheer size and complexity of the airline crew rostering problem, the crew rostering software adopted by airlines often take several hours to solve. Since a thorough scenario analysis may require constructing hundreds of different rosters, performing scenario analyses using commercial crew rostering software is considered too time-consuming to be applied in practice. Therefore, airlines could benefit from a supplementary scenario analysis tool, providing detailed insights on the effect of different crew rostering conditions on the constructed crew rosters.

This paper presents the development of a simulation model that can serve as a scenario analysis tool for airline cockpit crew rostering. The paper is structured as follows. First, Section II provides background information on the airline crew rostering problem. Subsequently, Section III discusses the development of the simulation model. Section IV explains the solution technique that is used in the simulation model. Section V presents the results of the performance analysis of the simulation model and of two scenario analyses that have been performed with the developed simulation model. Finally, Section VI reflects on the contributions and the applicability of the simulation model. Additionally, recommendations for future work are provided.

II. BACKGROUND

This section provides background information on the airline crew rostering problem, on different approaches to the problem and on the various solution techniques used to solve the problem.

A. Crew Rostering Problem

The airline crew rostering problem is a combinatorial optimization problem, and belongs to the class of NP-complete problems (Deveci & Demirel, 2018) and therefore it is also an NP-hard problem (Lucic & Teodorovic, 2007). The airline crew rostering problem is a *generalized assignment problem*, which is a special type of the *minimum cost multi-commodity flow problem* (Hillier & Lieberman, 2015). The minimum cost flow problem is the most general network model and encompasses many other specific problems such as the *shortest path problem*, the *minimum spanning tree problem* and the *assignment problem* (Burke & Kendall, 2005). The minimum cost flow problem considers flow through a network with limited arc capacities (Hillier & Lieberman, 2015). The objective in the minimum cost flow problem is to minimize the total cost of sending the available commodities through the network to satisfy the given demand (Hillier & Lieberman, 2015). The airline crew rostering problem is a multi-commodity flow problem where each crew member corresponds to a commodity and the demand is determined by the flight schedule (Cappanera & Gallo, 2004).

B. Crew Rostering Approaches

Outside of North America, *personalized rostering* is the most common approach for crew rostering (Kasirzadeh et al., 2017). With *personalized rostering* approaches, personalized rosters are constructed directly for each individual crew member, considering pre-assigned personal activities such as training days and vacations (Kasirzadeh et al., 2017). The more advanced personalized rostering systems also take into account individual crew preferences on certain activities or roster attributes as soft constraints in the roster optimization (Kohl & Karisch, 2004).

The *bidlines* approach is the approach that used to be common practice in North American airlines. In the *bidlines* approach, a set of anonymous monthly rosters (*bidlines*) are constructed, whereafter crew members place bids on the rosters they prefer (Jarrah & Diamond, 1997). Subsequently, the rosters are assigned to specific crew members based on seniority, meaning that if two or more crew members have placed a bid on the same roster, the roster will be assigned to the most senior crew member (Kohl & Karisch, 2004). After assignment of the bidlines, numerous pairings conflict with the individual pre-assigned activities of crew members such as training activities or granted days-off. Those pairings that are in conflict with pre-assigned activities are then removed from these bidlines. The least senior crew members will then be assigned to a combination of 'open pairings', i.e. pairings not covered by any of the generated bidlines (including those due to conflicts with pre-assigned activities), and reserve days (Campbell, Durfee, & Hines, 1997).

Nowadays, some large North American airlines have adopted the *Preferential Bidding System (PBS)* as their crew rostering approach (Achour, Gamache, Soumis, & Desaulniers, 2007; Kasirzadeh et al., 2017). The *Preferential Bidding System* is a special case of personalized rostering where the satisfaction of the preferences of senior crew members is prioritized over the satisfaction of those of junior crew members.

C. Crew Rostering Solution Techniques

There are two main groups of solution methodologies to the crew rostering problem: heuristics and mathematical programming methods (Bergh, Belin, de Bruecker, Demeulemeester, & de Boeck, 2013). However, certain researchers adopt hybrid methods, combining mathematical programming techniques with (meta-)heuristics to solve the problem.

Mathematical programming techniques are used to find the optimal solution to a given optimization problem. Mathematical programming methods encountered in airline crew rostering literature are based on integer programming and formulate the crew rostering problem as a set-partitioning problem (Deveci & Demirel, 2018; Kasirzadeh et al., 2017). In the set-partitioning problem formulation, the goal is to generate a roster for each crew member such that all pairings that need to be flown are covered (Kasirzadeh et al., 2017). Mathematical programming methods then may exploit various integer programming techniques such as *branch-and-bound* and *column generation* to solve the set-partitioning problem. An early influential crew rostering problem that exploited column generation techniques was proposed by M. Gamache, F. Soumis and G. Marquis (1999). A more recent work that is based on similar mathematical programming techniques is presented by A. Kasirzadeh, M. Saddoune and F. Soumis (2017).

Heuristics on the other hand aim to find a feasible suboptimal solution in reasonable computational time (Burke & Kendall, 2005). Problem specific knowledge is exploited in order to define a set of rules that need to be followed to construct a good feasible solution. In academic literature, heuristics are often divided in *constructive heuristics* and *improvement heuristics* (Bergh et al., 2013; Ernst, Jiang, Krishnamoorthy, Owens, & Sier, 2004). R.D. Jones (1989) presented an early heurstic procedure to solve the rostering problem. First an initial roster is constructed, whereafter moving and swapping opportunities are used to improve the initial solution. Recent works on crew rostering heuristics involve models that are based on *meta-heuristics* (Lucic & Teodorovic, 2007; Maenhout & Vanhoucke, 2010) or *matheuristics* (Doi, Nishi, & Voß, 2018), which can be considered special types of techniques that combine different basic heuristics in order to explore the search space more efficiently (Blum & Roli, 2001; Burke & Kendall, 2005).

III. MODEL DEVELOPMENT

This section discusses the development of the simulation model. First, Section III-A discusses the scope of the rostering problems for which the simulation model is developed. In Section III-B, the mathematical model formulation of the crew rostering problem is presented. Section III-C then presents an overview of the main components of the simulation model.

A. Scope

The simulation model is developed in collaboration with a major European airline, leveraging actual historical pairing, crew and roster data in order to accurately model the cockpit crew rostering process. This section discusses the crew rostering approach, the main assumptions and the rostering objective considered in the development of the simulation model.

1) Approach: The developed simulation model addresses the personalized crew rostering problem with a weekly rolling roster period of four weeks. The first three weeks of the monthly roster are fixed and every week a new fourth week is added to the monthly roster. Every week, an updated roster for the coming four weeks is published to the pilots. For example, at time t in *Week 1* the roster for *Week 5* is constructed, added to the previous published roster and subsequently published to the pilots. Figure 1 illustrates the evolution of the published roster over time.

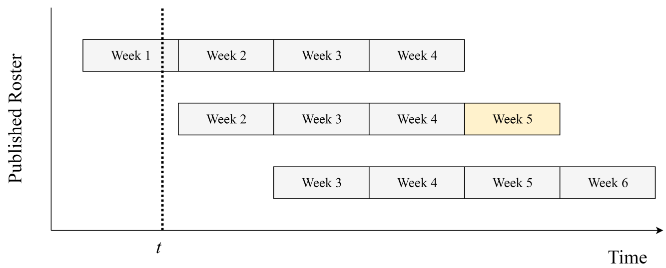

Fig. 1. Published Roster over Time using a Rolling Horizon

In the rolling horizon approach, an airline has to construct a weekly roster that covers all pairings and reserve blocks that start in the fourth week of the roster. In personalized crew rostering, the rostering of pairings is done under the consideration of personal pre-assigned activities of pilots. Certain personal activities are rostered with a longer roster horizon than pairings, resulting in that at the moment of pairing rostering these personal activities are already pre-assigned. Pre-assigned activities may include granted flight requests, training activities and vacations. For example, training activities might be rostered up to 10 weeks in advance and vacation scheduling might be done up to 6 months in advance.

Figure 2 illustrates an example crew rostering problem instance involving five pilots and four pairings. In this research, a pairing block consists of days of flying (F) and days off, so-called *travel leave* (TL). Flying days include days of actual flying, i.e. departure from and return to the crew base, but also days spent on destination in between. Pre-assigned activities include pairings rostered in the previous roster period, requested flights (RF), requested leave (RL), training activities (SIM), vacations (V), medical checks (M) and office days (O).

Pre-Assigned Activities

Set of Pilots	Weeks	5							6							7				
	Days	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Pilot001		TL	TL	TL															SIM	
Pilot002		RF	RF	TL	TL	TL	RF	RF	RF	RF	TL	TL	TL	V	V	V	V	V	V	V
Pilot003																		RF	RF	RF
Pilot004		TL	TL									SIM								
Pilot005						RL	RL													

Set of Pairings

P001		F	F	F	F	TL	TL	TL			
P002			F	F	F	TL	TL	TL	TL		
P003					F	F	F	TL	TL		
P004					F	F	F	F	TL	TL	TL

Fig. 2. Example Crew Rostering Problem Instance

2) Assumptions: The main assumptions made in the development of the simulation model are the following:

- Only one crew base is considered.
- Only one rank per problem instance is considered; '*flying below rank*' is not allowed.
- The length of travel leave in a pairing block is adjusted for part-time pilots.
- There is no constraint on maximum flying hours per pilot.
- Crew preferences are taken into account as pre-assigned flight requests and leave requests.
- Route instruction flights are considered pre-assigned.
- Pre-assigned activities are fixed and cannot be rescheduled or canceled.
- Roster efficiency is the only objective in the rostering process.

3) Objective: For a given set of pairings, pilots and pre-assigned activities, the roster inefficiency should be minimized. In this research, roster inefficiency is defined as unused labor potential and is expressed in number of '*premium days*'. Premium days are considered a currency between an airline and its crew members. Crew members can earn premium days by performing additional tasks for the airline, such as office tasks or doing overtime. Earned premium days are added to the '*premium days budget*' of the pilot and are remunerated to crew members as additional days off somewhere in the future. The objective of the airline is twofold: the build-up of premium days should be minimized and the build-off of premium days should be maximized.

B. Mathematical Model

The weekly personalized crew rostering problem is modeled as a set-partitioning problem where the objective is to find a roster for each pilot such that all pairings are covered at minimum cost. The objective function minimizes the sum of the cost of all individual rosters that are assigned to pilots. The cost of an assigned roster is a linear combination of four cost parameters, which are expressed in number of days of inefficiency. The following cost parameters are included in the model:

- **Lost days**: Rosters containing '*open days*', i.e days without an activity, can be used to allocate premium days off. However, if the considered pilot has a zero premium days budget, the open day in its roster becomes a '*lost day*'. A lost day is a day of inefficiency as it corresponds to a missed opportunity to allocate a premium day off.

- **Flexible Travel Leave (FTL)**: The last day of the travel leave in a pairing block, i.e. the '*flexible travel leave*' (FTL), can be removed from the pairing in order to free up space in the roster. However, this comes at a cost since pilots are compensated for the removal of their travel leave in premium days. The compensation in premium days corresponds to days of inefficiency since these premium days need to be allocated as extra days off in future rosters.

- **Next week penalty**: The objective function is penalized for rosters that are expected to cause inefficiencies in subsequent roster periods. Whether an individual roster is expected to cause inefficiency is evaluated according to the resulting gap in next week's roster, which spans up to the next pre-assigned activity. If this gap is below a certain threshold, it is expected that it will be hard to assign a subsequent pairing in that gap in next week's roster period, causing inefficiency. In the simulation model, this threshold value depends on the length of the pairings in the pairing set and is adjusted for part-time pilots.

- **Uncovered pairings**: Pairings or reserve blocks that are not assigned to any pilot become *uncovered*. Uncovered pairings need to be recovered by convincing pilots to do overtime, for which pilots receive compensation in premium days. In the simulation model, uncovered pairings get assigned to a fictitious ('*slack*') pilot in order to capture this cost parameter in the objective function.

Below, the mathematical model formulation of the set-partitioning problem can be found.

Sets

P	Set of all pairings to be covered
M	Set of all available pilots
R_m	Set of all possible rosters for pilot m
s	The slack pilot
R_s	Set of all possible roster for slack pilot s

Parameters

$c^{ld}_{r,m}$	Lost days cost if roster $r \in R$ is assigned to pilot m
$c^{ftl}_{r,m}$	Flexible Travel Leave cost if roster $r \in R$ is assigned to pilot m
$c^{nw}_{r,m}$	Penalty cost for next week's roster if roster r is assigned to pilot m
$c^{u}_{r,s}$	Uncovered pairing cost if roster $r \in R$ is assigned to slack pilot s
l_p	Length of pairing $p \in P$ for a full-time pilot

$$y_{p,r} = \begin{cases} 1 & \text{if pairing } p \text{ is covered in roster } r \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Variables

$$x_{r,m} = \begin{cases} 1 & \text{if roster } r \text{ is assigned to pilot } m \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$x_{r,s} = \begin{cases} 1 & \text{if roster } r \text{ is assigned to slack pilot } s \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Model formulation:

$$min \sum_{m \in M} \sum_{r \in R_m} (c^{ftl}_{r,m} + c^{ld}_{r,m} + c^{nw}_{r,m}) \cdot x_{r,m} + \sum_{r \in R_s} c^{u}_{r,s} \cdot x_{r,s} \tag{4}$$

Subject to:

$$\sum_{m \in M} \sum_{r \in R_m} y_{p,r} \cdot x_{r,m} + \sum_{r \in R_s} y_{p,r} \cdot x_{r,s} = 1, \qquad \forall p \in P \tag{5}$$

$$\sum_{r \in R_m} x_{r,m} = 1, \qquad \forall m \in M \tag{6}$$

Equation 4 presents the objective function used in the simulation model. Below, in Equation 7, the incorporation of the costs due to uncovered pairings is clarified. The number of premium days compensation required to convince a pilot to do overtime in order to recover an uncovered pairing is approximated by the length of the uncovered pairing. The value of this cost parameter has been determined in consultation with operators of the collaborating airline and is considered a close representation of the inefficiency costs resulting from uncovered pairings.

$$c^{u}_{r,s} = \sum_{p \in P} l_p \cdot y_{p,r} \tag{7}$$

The set-partitioning problem formulation involves two constraints: each pairing should be covered exactly once, either by an actual pilot or by the slack pilot, (5) and each pilot must have exactly one roster (6).

C. Computational Model

The simulation model is developed using Python 3.6. Figure 3 presents a high-level overview of the components of the simulation model. This section briefly addresses the individual blocks of the simulation model. Afterwards, in Section IV the developed solution technique is discussed into more detail.

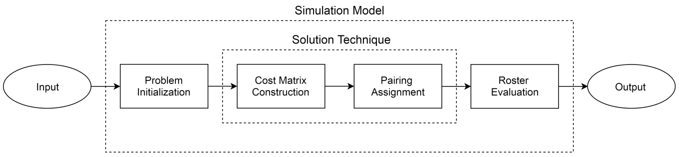

Fig. 3. Overview of the Computational Model

1) Problem Initialization: The model takes historical '*published rosters*' as input. From the published roster, the roster as it was just before pairing assignment, i.e. the '*prefixed roster*', and the set of pairings and reserves that should be rostered in the considered week are obtained. Figure 4 presents the block diagram of the problem initialization module.

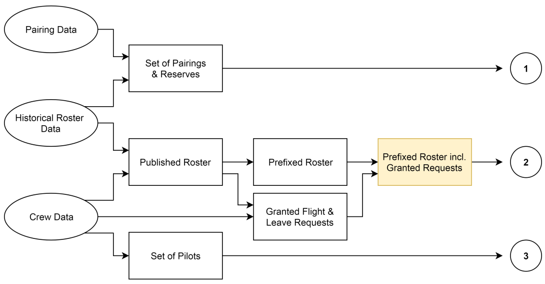

Fig. 4. Module 1: Problem Initialization

2) Cost Matrix Construction: The cost matrix maps the different cost parameters of each pilot-roster assignment combination. Figure 5 presents the block diagram of the cost matrix construction module. The construction of the cost matrix is explained in Section IV.

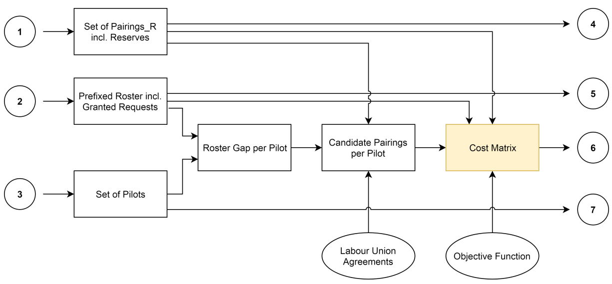

Fig. 5. Module 2: Cost Matrix Construction

3) Pairing Assignment: Pairing assignment can be done in various different ways, using different assignment algorithms. Figure 6 depicts a constructive pairing assignment algorithm, where the roster is gradually constructed through the subsequent assignment of pairings to pilots. The specific pairing assignment algorithm used in the simulation model is discussed in detail in Section IV.

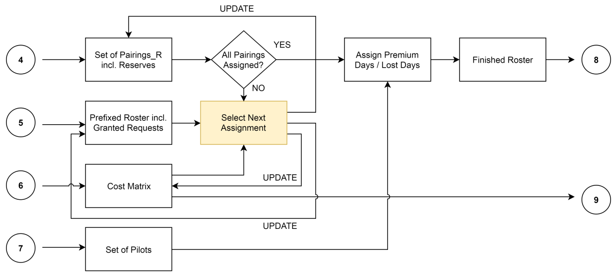

Fig. 6. Module 3: Pairing Assignment

4) Roster Evaluation: After all pairings have been assigned, the finished roster is evaluated and exported to Microsoft Excel. To compare different pairing assignment algorithms, the simulation model allows for the evaluation and processing of multiple finished rosters. Figure 7 depicts the evaluation module.

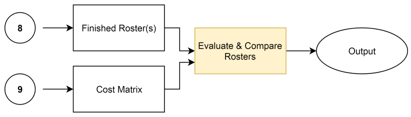

Fig. 7. Module 4: Roster Evaluation

IV. SOLUTION TECHNIQUE

The simulation model solves the formulated set-partitioning problem using a novel solution method that is based on the Hungarian assignment algorithm. This section explains the novel solution method, called the '*Hungarian Improvement*' algorithm. First, the Hungarian assignment algorithm on which the developed solution method is based, is explained. Afterwards, the Hungarian Improvement algorithm is described.

A. Hungarian Assignment Algorithm

The Hungarian assignment algorithm is a classic assignment algorithm that solves the weighted '*bipartite*' matching problem. The bipartite matching problem is the problem of assigning '*jobs*' to '*workers*', where each job can be assigned to at most one worker and every worker can have at most one job assigned. When an individual cost is associated with each possible assignment, the bipartite matching problem becomes a '*weighted*' bipartite matching problem (Frank, 2005).

1) Bipartite Reduction: For the Hungarian assignment algorithm to be applied on the airline crew rostering problem, the set-partitioning problem is reduced to a minimum-cost weighted bipartite matching problem. In the reduced problem, the set of rosters R_m for each pilot is reduced to contain only rosters that contain just a single pairing. The bipartite matching problem formulation then implies that each pairing needs to be assigned to one pilot and that each pilot can have at most one pairing assigned.

2) Cost Matrix Construction: The Hungarian assignment algorithm requires the construction of the bipartite cost matrix that maps all costs of the possible pilot-pairing combinations. In the developed simulation model, the cost matrix is computed by evaluating the four different cost parameters in the objective function for all possible pilot-pairing combinations.

To increase the computational efficiency of the cost matrix construction process, first for every pilot the current '*gap*' in the roster of the pilot is calculated, which spans from the end of the last activity rostered in the previous roster period to the first upcoming pre-assigned activity. Subsequently, based on the start date and length of the gap, the set of *candidate* pairings for every pilot is determined. Candidate pairings that violate rostering constraints stemming from labour union agreements are removed from the set of candidate pairings. Consequently, only for these candidate pairings the costs need to be calculated. All other pilot-pairing combinations are infeasible, resulting in a *Big M* in the constructed cost matrix.

3) Addition of Slack: To capture the costs of uncovered pairings, a slack pilot is included in the cost matrix that maps the uncovered pairing cost for all considered pairings. Also, slack pairings that corresponds to an empty roster is added to the cost matrix in order to account for the roster inefficiency costs of empty rosters. These empty roster costs are pilot-dependent since the 'lost days' and 'next week penalty' cost parameters depend on the premium days budget and pre-assigned activities of a pilot.

The number of slack pairings added to the cost matrix is determined by computing the difference between the number of pilots and the number of pairings in the considered problem. The problem then is solved using the Hungarian assignment algorithm. If the minimum-cost bipartite solution involves uncovered pairings, a number of slack pilots corresponding to the number of uncovered pairings is added to the cost matrix. Subsequently, the same number of slack pairings is added to the matrix, with a Big M cost for the entries corresponding to an assignment of a slack pairing to a slack pilot. Afterwards, the weighted bipartite problem is solved again in order to compute the minimum-cost bipartite solution to the given crew rostering problem.

4) Solution Computation: The Hungarian assignment algorithm solves the bipartite matching problem by performing matrix operations (Munkres, 1957). The Hungarian assignment algorithm first performs row reduction and column reduction operations on the input cost matrix. The optimal set of assignments is attained if the minimum number of lines to cover all zero elements in the resulting cost matrix equals the number of rows (Hillier & Lieberman, 2015). If the minimum number of lines to cover all zeros is less than the number of rows, the following steps are performed iteratively until an optimal set of assignments is obtained (Hillier & Lieberman, 2015):

- Cover all zeros in the reduced cost matrix using the minimum possible number of horizontal and vertical lines;
- Subtract the smallest uncovered number from every other uncovered number;
- Add the smallest uncovered number to the numbers at intersections of covering lines;
- Numbers crossed out but not at the intersections of cross-out lines carry over unchanged to the next table.

B. Hungarian Improvement Algorithm

The Hungarian assignment algorithm is a very efficient solution method for the bipartite matching problem. However, in the airline crew rostering problem pilots can have two pairings and with very short pairings even three pairings assigned in a single weekly roster period. Therefore, a novel solution method has been developed that is based on the Hungarian assignment algorithm, called the '*Hungarian Improvement*' algorithm. The Hungarian Improvement algorithm exploits the resemblance of the weekly crew rostering problem to a minimum-cost weighted bipartite matching problem. This Subsection presents the characteristics of the Hungarian Improvement algorithm.

1) Pairing Assignment Reformulation: In the Hungarian Improvement algorithm two different cost matrices are constructed, the first of which is the bipartite cost matrix as explained in the previous subsection. The second cost matrix is based on a reformulation of the set-partitioning roster assignment problem to a pairing assignment problem.

In the pairing assignment reformulation, the decision variables do not correspond to the assignment of a roster to a pilot, but to the assignment of a single pairing to a pilot. In the pairing assignment reformulation, the costs of a pairing assignment are calculated taking into consideration that a subsequent pairing can be assigned to the same pilot in the same roster period. This reformulation changes the 'lost days' cost parameter of the objective function. Also, the 'next week penalty' cost parameter changes and is referred to as the 'next assignment penalty' in the reformulation. The assignment reformulation of the crew rostering problem reformulates the set-partitioning problem as an integer programming formulation where each pairing needs to be assigned to exactly one pilot.

2) Roster Construction: The developed Hungarian Improvement algorithm can be compared to a greedy tree search improvement heuristic. The initial solution in the tree search is the minimum-cost bipartite solution, which is calculated using the Hungarian assignment algorithm as explained in Subsection IV-A. The Hungarian Improvement algorithm then aims to improve the initial solution by identifying specific pilot-pairing assignments with an improved *remaining bipartite solution*. The remaining bipartite solution is defined as the solution to the crew rostering problem if the remaining pairings after this specific pilot-pairing assignment are assigned using a minimum-cost weighted bipartite matching.

The Hungarian improvement algorithm is an iterative procedure where in every step the pilot-pairing assignment is selected that is associated with the highest improvement in remaining bipartite solution compared to the bipartite solution in the previous iteration. If no pilot-pairing assignments with an improved remaining bipartite solution are found, the iterative search for Hungarian improving assignments stops and the remaining pairings are assigned according to the remaining bipartite solution as computed by the Hungarian assignment algorithm.

Below, the pseudo-code of the Hungarian Improvement algorithm can be found. The search for Hungarian improving assignments is further explained in the next subsection.

Algorithm 1: Roster Construction using the *Hungarian Improvement* Algorithm

```
Data: Pilots, pairings, prefixed roster, bipartite cost
      matrix, assignment cost matrix, tree-parameter
Result: Roster with pairings assigned and premium
        days off allocated
initialization ;
while True do
    Algorithm 2: search for Hungarian improving
      assignments ;
    if no Hungarian improving assignment found then
        break
    else
        select Hungarian improving assignment with
          greatest improvement ;
        assign selected pairing to selected pilot ;
        update cost matrices of selected pilot ;
        remove selected pairing from cost matrices ;
    end
end
solve remaining bipartite cost matrix using Hungarian
  assignment algorithm ;
for assignments ∈ remaining bipartite solution do
    select assignment ;
    assign selected pairing to selected pilot ;
end
allocate premium days or add lost days;
return Finished roster
```

3) Search for Hungarian improving assignments: In the search for '*Hungarian improving assignments*', nodes correspond to individual pilot-pairing assignments. Each node is evaluated on the remaining bipartite solution plus the assignment cost of the corresponding pilot-pairing assignment. The pairing assignment reformulation as explained in the previous subsection is used in the search for Hungarian improving assignments to calculate the costs of the individual pilot-pairing assignments. Also, each node has a remaining bipartite cost matrix associated to it that maps the remaining weighted bipartite matching problem if the corresponding pilot-pairing assignment is selected. During the traversal of the tree search, the bipartite and assignment cost matrices are updated dynamically in order to account for changing costs due to pairings that are assigned in previous iterations.

The '*tree-parameter*' governs the number of nodes, i.e. pilot-pairing combinations, that is considered during each iteration. The larger this tree-parameter is, the more pilot-pairing assignments are evaluated in each iteration of the greedy tree search. A larger tree-parameter thus results in crew rosters of potentially higher solution quality at the cost of computational time.

By adjusting the tree-parameter value in the Hungarian Improvement algorithm, multiple configurations of the developed simulation model can be constructed. The simulation model therefore is considered '*adaptable*', being able to adapt its performance in terms of solution quality and computational time to the needs of the operator.

Below, the pseudo-code of the search for Hungarian improving assignments can be found.

Algorithm 2: Search for Hungarian Improving Assignments

```
Data: Pilots, pairings, prefixed roster, bipartite cost
      matrix, assignment cost matrix, tree-parameter
Result: Dictionary with Hungarian improving
        assignments
initialization;
calculate initial remaining bipartite solution using
  Hungarian method ;
determine set of 'pilot options' that can have multiple
  pairings assigned ;
for pilots ∈ pilot options do
    determine set of 'pairing options' that can be
      assigned ;
    select a number of pairing options to be evaluated,
      using the 'tree-parameter';
    for pairings ∈ selected pairings do
        temporarily assign selected pairing to selected
          pilot ;
        temporarily update bipartite cost matrix ;
        solve remaining bipartite cost matrix using
          Hungarian assignment algorithm ;
        calculate improvement of cost of selected
          pairing assignment and remaining bipartite
          solution over cost of initial remaining bipartite
          solution;
        if improvement > 0 then
            save Hungarian improving assignment to
              dictionary
        end
    end
end
return Dictionary with Hungarian improving
  assignments
```

4) Graphic Representation: Figures 8 and 9 provide a graphical representation of the Hungarian Improvement algorithm. The nodes ($H0, H1, H2$ ... Hn) correspond to specific pilot-pairing assignments, where the value of n depends on the input *tree-parameter*. Each node has a cost matrix associated to it, representing the cost matrix of the remaining rostering problem. Using this cost matrix, the remaining bipartite solution for each node is calculated.

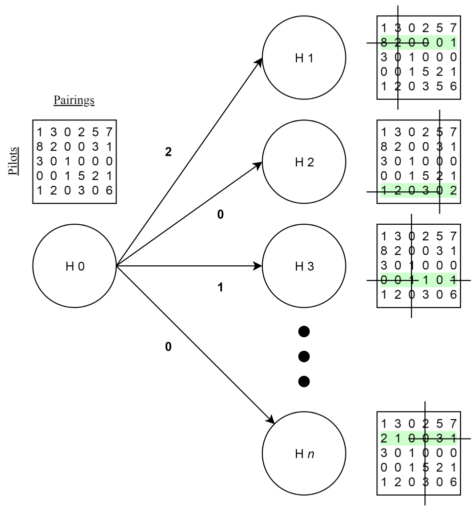

Fig. 8. The first iteration of the Hungarian Improvement Algorithm

The pilot-pairing assignment with the lowest sum of remaining bipartite solution cost and pilot-pairing assignment cost (depicted along the arrows in the figures) is selected. Figure 9 shows the next iteration in the Hungarian Improvement procedure after one of the pilot-pairing assignments ($H2$) has been chosen. If none of the nodes provide an improvement in total cost, the improvement procedure is stopped and the remaining bipartite solution is accepted for the assignment of the remaining pairings.

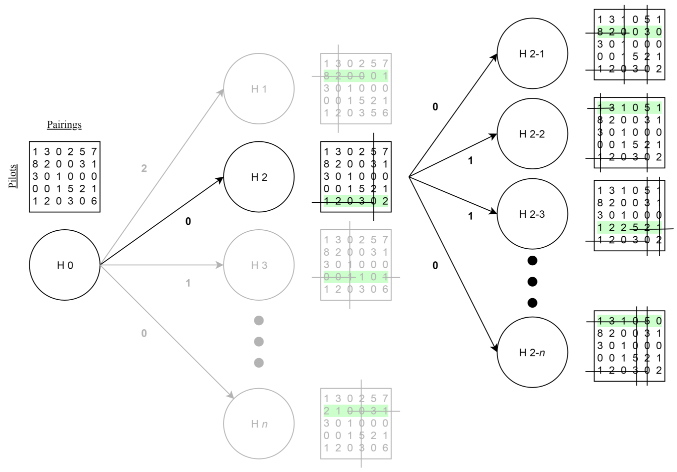

Fig. 9. The second iteration of the Hungarian Improvement Algorithm

V. Experiments

The developed simulation model has been tested in three experiments. The first experiment evaluates the performance of the simulation model and compares it to two benchmark models. Afterwards, two experiments have been performed that test the applicability of the simulation model as a scenario analysis tool. This section presents the results of the performed experiments.

A. Performance Analysis

The performance of various configurations of the developed simulation model has been assessed and compared to two benchmarks: a model based on the Hungarian assignment algorithm and a model based on linear programming. The experiments have been conducted on a HP Elitebook 8570w with an Intel(R) Core(TM) processor running at a rate of 2.40 GHz. This subsection first introduces the benchmark algorithms. Subsequently, the performance of the simulation model is compared and assessed on solution quality versus computational time. Afterwards, the sensitivity of the simulation model to problems of increasing size is assessed and compared to the benchmark algorithms.

1) Benchmark Algorithms: The simulation model is compared to two benchmark algorithms: the Hungarian assignment algorithm and a linear programming (LP) algorithm.

- **Hungarian assignment algorithm (H):** The Hungarian assignment algorithm can be considered the baseline configuration of the developed Hungarian Improvement algorithm with a tree-parameter of 0. The Hungarian assignment algorithm finds the minimum-cost bipartite solution to a given crew rostering problem. The characteristics of the Hungarian assignment algorithm are explained in Subsection IV-A.

- **Linear Programming algorithm (LP):** The linear programming model solves the complete set-partitioning problem as formulated in Subsection III-B. The bipartite cost matrix, as explained in Section IV, is used to identify which combinations of pairings need to be added as rosters to the bipartite cost matrix to form the '*complete cost matrix*'. Subsequently, the set of all possible rosters are added as decision variables to the LP model. The cost parameters in the objective function then follow from the complete cost matrix. Finally, constraints on pairing coverage and pilot availability constraints are added to the model.
The resulting LP model is solved using version 8.1 of the Gurobi Optimizer. The solution produced by the LP model serves as the reference for solution quality and the solution quality of the simulation model is assessed in terms of percentage gap from this optimal solution.

2) Solution Quality vs. Computational Time: The performance of the simulation model has been tested using a set of 44 problem instances, corresponding to 22 weeks of the Captain (CP) and First Officer (FO) rosters of a small long-haul aircraft division. The problems of the considered crew divisions involve around 150 pilots and 100 pairings. The simulation model has been tested using various configuration settings, corresponding to different values of the tree-parameter as explained in Subsection IV.

Rosters produced using the various configurations of the simulation model are evaluated and compared to the rosters that are produced by the benchmark models. Solution quality is expressed in percentage from the optimal solution as produced by the LP model. Figure 10 presents the average solution quality obtained for various configurations of the simulation model against the average required computational time. Configurations of the simulation model are depicted by a C and the corresponding tree-parameter (1, 2, 5 etc.). The $C\infty$ configuration is the simulation model with the Hungarian Improvement algorithm that considers all pilot-pairing options in the search for Hungarian improving assignments.

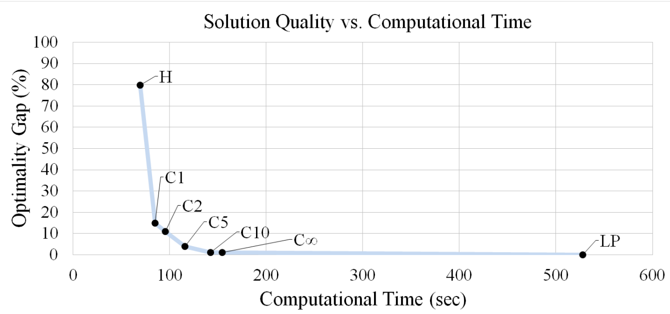

Fig. 10. Performance of various simulation model configurations

Below, in Table I a more detailed overview of the performance of the simulation model configurations is provided. From the performance analysis, it can be concluded that on average the developed Hungarian Improvement algorithm constructs rosters of near-optimal solution quality in around just 20% of the computational time required by the LP algorithm. Also, the performance analysis confirms that increasing the tree-parameter in the Hungarian Improvement algorithm improves the solution quality of the constructed roster at the expense of computational time.

TABLE I

PERFORMANCE OF VARIOUS SIMULATION MODEL CONFIGURATIONS

	Optimality Gap (%)		Computational Time (sec)	
	Mean	Std. Dev.	Mean	Std. Dev.
H	79,8	106	69,3	20,3
C1	15,0	32,2	84,6	30,7
C2	10,9	31,5	95,6	37,8
C5	3,95	15,7	116	51,9
C10	1,02	4,72	142	80,5
C∞	1,02	4,72	154	98,5
LP	0	0	528	268

3) Computational Time vs. Problem Size: The scalability of the simulation model has been tested using a set of 10 problem instances, involving around 450 pilots and 270 pairings. From these problem instances, artificial problem instances have been constructed by considering parts of the set of pilots, pairings and pre-assigned activities and taking these artificially created roster conditions as input. By doing so, problem instances of varying problem sizes have been created. The scalability of various configurations of the simulation model have been tested and compared to the two benchmarks algorithms (H and LP). Figure 11 presents the average computational time required for the different simulation model configurations to solve problems of varying problem sizes.

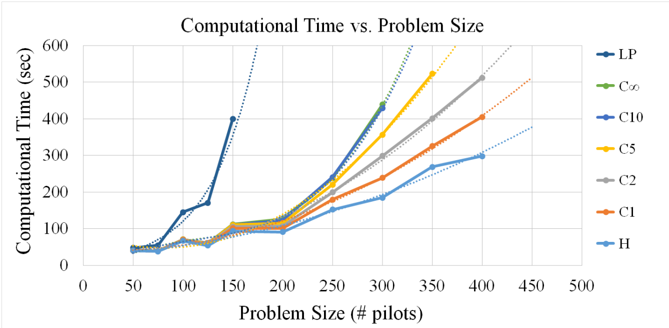

Fig. 11. Computational time growth per simulation model configuration

The results show that the computational time required by the simulation model is less sensitive to increases in problem size compared to the LP algorithm. Whereas the LP algorithm seems to be increasing exponentially with increasing problem size, the simulation model configurations seem to have a quadratic order of growth. Furthermore, the computational time growth of the simulation model depends on the configuration of the simulation model. This characteristic makes the developed simulation model specifically suitable for its purpose: to serve as a scenario testing tool. Depending on the design and requirements of a scenario analysis, a suitable configuration of the simulation model can be selected.

A specific scenario analysis may involve the construction of multiple rosters using the same input roster conditions. In such an analysis, the problem can be initialized once before performing multiple runs on the same initialized problem. Therefore, the growth of the required pre- and postprocessing time and that of the actual computational time required for solving the problem, have also been assessed separately. Figures 12 and 13, the growth of the computational time for these different components of the simulation model are presented separately.

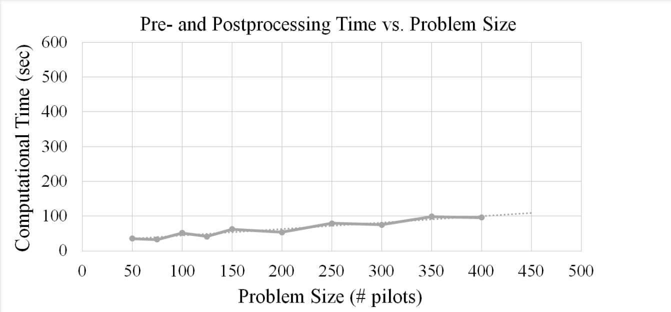

Fig. 12. Pre- and postprocessing time growth of the simulation model

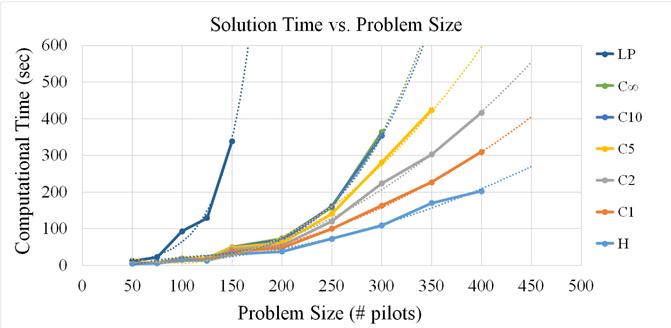

Fig. 13. Solution time growth of various simulation model configurations

B. *Scenario Analysis I: Flexible Travel Leave*

In tight and difficult rosters, the considered airline has the possibility to remove the last day of certain pairings, in order to free up space in the roster. This last day of the pairing block is called the '*Flexible Travel Leave* (FTL)'. To compensate for the removal of the FTL, pilots receive this removed rest day as a day off somewhere in the coming weeks plus an extra day off as compensation.

Although the removal of FTL comes at a cost in terms of compensated days off, the additional flexibility as a result of the possibility to remove FTL is expected to increase the roster efficiency. The experiment that has been performed assesses the benefit of having pilots that allow for the removal of their FTL, from here on called '*FTL pilots*'.

1) Scenario Analysis Design: The FTL experiment has been performed on 44 problem instances corresponding to 22 different weeks of the Captain (CP) and First Officer (FO) rosters of a small long-haul aircraft division. The problem instances involve around 150 pilots and 100 pairings.

The simulation model was used to test the solution quality of the produced rosters using varying numbers of FTL pilots. The number of FTL pilots was varied in steps of 5%, ranging from having no FTL pilots to being able to remove the FTL for every pilot. In each run, based on the input FTL percentage, random pilots were selected as FTL pilots.

With 44 problem instances and 21 runs per problem instance, a total of 924 rosters were to be constructed in the scenario analysis. For the total scenario analysis a requirement on computational time was set to be a maximum of 16 hours, corresponding to running the experiment for the two different ranks, CP and FO, separately overnight.

2) Configuration Selection: The scenario analysis as explained in the previous subsection required the construction of 924 rosters in 16 hours for problem instances with an average size of around 150 pilots. Pre-processing, taking on average around 60 seconds for problems with 150 pilots, only had to be done once for every of the 44 considered roster weeks. This left around 15 hours of actual solution time for the construction of 924 rosters, resulting in a maximum average solution time of around 60 seconds for roster construction. Adding the 60 seconds of pre-processing, a maximum of 120 seconds of average solution time per constructed roster was set for the scenario analysis.

From the performance analysis as discussed in Subsection V-A.2 and presented in Table I, the best performing simulation model configuration for this problem size within the maximum computational time is the C5 configuration. Therefore, the C5 configuration of the simulation model has been selected for the FTL scenario analysis experiment.

3) Results: For a consistent evaluation of the benefit of having FTL pilots over varying roster weeks, the results are normalized. For every week, the objective function value of the constructed roster using 100% FTL pilots was taken as the reference. For every roster week, 21 rosters have been constructed by varying the number of FTL pilots from 0 to 100% in steps of 5%. The objective function value of the constructed rosters are normalized to the objective function value of the constructed roster using 100% FTL pilots. Figure 14 presents the results of the FTL scenario analysis experiment. The three blue lines in the graph depict the first quartile, median and third quartile value of the aggregated results. The orange line is the average normalized objective function value over varying percentages of FTL pilots.

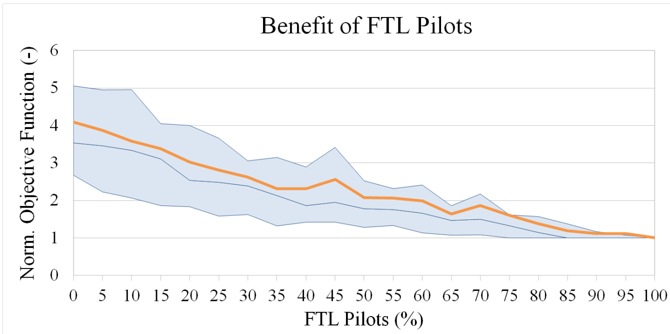

Fig. 14. The impact of having FTL pilots on roster efficiency (1)

The result shows that the number of FTL pilots has a major impact on the resulting efficiency of the constructed rosters. With the possibility to remove the FTL for all pilots, the inefficiency in number of days is four times as low as the rosters that are constructed without having the possibility to remove the FTL for any pilot.

The trend between the objective function value and the number of FTL pilots seems to be linear. However, from the graph it can be seen that especially the first 30% to 35% of FTL pilots are crucial, having a higher impact on the average solution quality of the constructed rosters as compared to adding FTL pilots above that threshold. Figure 15 presents the same graph as Figure 14 but showing two separate linear trends that intersect at 30% of FTL pilots. From this scenario analysis, a possible conclusion for an airline could be that a minimum of 30% of FTL pilots should be pursued.

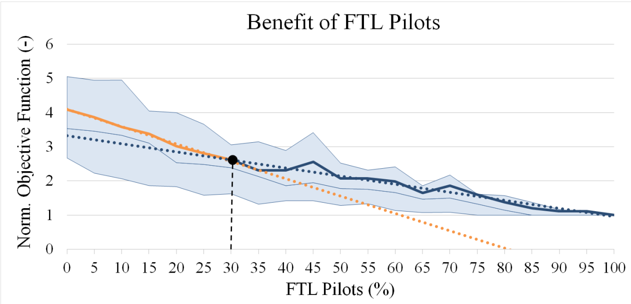

Fig. 15. The impact of having FTL pilots on roster efficiency (2)

C. Scenario Analysis II: Dual Pilots

The second scenario analysis focuses on the rostering problem of a '*combined aircraft division*'. A combined aircraft division is an aircraft division where pilots can be qualified to operate two different aircraft types. As a result, the crew rostering problems of these different aircraft types become an integrated rostering problem, where pilots that are qualified to fly on both aircraft types, from here on called *dual pilots*, are important resources. An example of a combined crew rostering problem could be the Airbus A330 and A350 rostering problems or the Boeing B777 and B787 rostering problems.

The aim of the second scenario analysis experiment is to evaluate the benefit of having dual pilots. Dual pilots provide airlines with the flexibility to spread the workload of two aircraft divisions over the combined group of pilots. Overall, this therefore leads to rosters with less inefficiency costs. However, this increase in roster flexibility comes at a cost since dual pilots need to be trained and need to stay current on both aircraft types. Therefore, in order to make a well-grounded trade-off between the benefits and costs of having dual pilots, it is important to have a clear evaluation of the benefit of dual pilots.

1) Scenario Analysis Design: The second scenario analysis experiment was performed on 44 problem instances corresponding to 22 different weeks of the Captain (CP) and First Officer (FO) rosters of a medium size combined long-haul aircraft division. The considered problem instances involved around 450 pilots and 270 pairings.

In the scenario analysis, pilots are either '*single pilots*' or '*dual pilots*'. Dual pilots can fly on both of the aircraft types and the benefit of having these dual pilots is assessed by constructing rosters for different percentages of dual pilots.

The number of dual pilots was varied in steps of 5%, ranging from having 30% to 60% dual pilots. Due to the large size of the considered problems, not the full range between 0% and 100% of dual pilots was assessed. The 30% and 60% limits were based on the pairing composition in the considered problem sets and the scenarios between 30% and 60% were considered to be the most interesting for evaluation.

In each run of the dual pilots scenario analysis, random pilots were selected as dual pilots. With 44 problem instances and 7 runs per problem instance, the dual pilots scenario analysis required the construction of 308 crew rosters for problems involving around 450 pilots.

2) Configuration Selection: The scenario analysis explained in the previous subsections requires the construction of 308 rosters for problem instances with an average size of around 450 pilots. From the performance analysis as discussed in Subsection V-A.3, the expected computational time is estimated for the various configurations of the simulation model.

Even for configurations of the simulation model with low tree-parameters, the total computational time of the scenario analysis is estimated to be over 30 hours. The Hungarian algorithm has a lower estimated total computational time of around 24 hours, but its inferior performance in terms of solution quality makes it less reliable as a scenario analysis model. Therefore, the configuration of the simulation model with the lowest tree-parameter, the C1 configuration, is selected for the considered scenario analysis.

3) Results: For a consistent evaluation of the benefit of dual pilots over varying roster weeks, the results are normalized. For every week, the objective function value of the constructed roster using 60% dual pilots was taken as the reference. Per week, the constructed rosters using other dual pilots percentages are normalized to the objective function value of the constructed roster using 60% dual pilots. Figure 16 presents the results of the dual pilots scenario anlaysis. The three blue lines in the graph depict the first quartile, median and third quartile value of the aggregated results. The orange line is the average normalized objective function value over varying percentages of dual pilots.

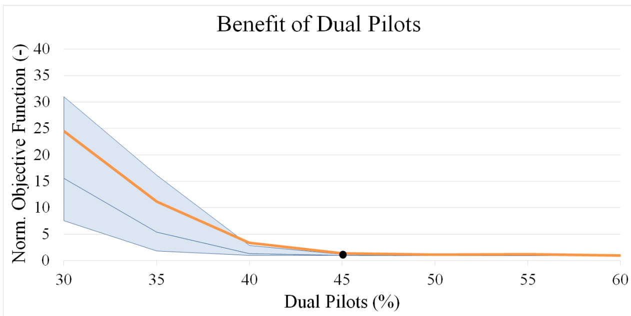

Fig. 16. Benefit of Dual Pilots

The performed scenario analysis indicates that a minimum of around 45% dual pilots is needed to construct rosters of acceptable solution quality. However, a detailed assessment of the results shows that even with 55% of dual pilots, the average inefficiency is almost 20% higher than the average inefficiency in rosters with 60% dual pilots. The outcome of this scenario analysis therefore suggests that a follow-up scenario analysis experiment should be performed, assessing the impact on roster efficiency if the percentage of dual pilots is increased above 60%. However, the results of this initial scenario analysis experiment indicate that with more extensive scenario analysis testing, the benefit of having dual pilots can be quantified.

VI. DISCUSSION

The purpose of the simulation model was to serve as a scenario analysis tool, providing detailed insights in the effect of changing crew rostering conditions on the constructed crew rosters. The developed solution technique has been found to be particularly suitable for the purpose of scenario analysis. Depending on the design and requirements of scenario analysis experiments, a suitable configuration of the simulation model can be selected, allowing airline operators to make the trade-off between scenario analysis accuracy and computational time. The viability of the simulation model as a scenario analysis tool has been tested by means of two initial scenario analysis experiments. The two experiments have confirmed the practical value of the simulation model, proving it to be capable of constructing and evaluating hundreds of different rosters in limited time.

Besides the development of the simulation model itself, an important contribution of this research is the problem definition. From extensive literature study it was concluded that there is a mismatch between the focus of academic literature and industry needs. While academic literature on airline crew rostering focuses on developing more efficient solution methods for the crew rostering problem, the main challenge at airlines is to determine the appropriate set of pilots, pairings and rostering policies that allow for an efficient and favorable operation. Without a thorough understanding of the impact of the changes in crew composition, flight schedule and labour union agreements, airlines run the risk of agreeing to long-term commitments that significantly deteriorate the roster efficiency.

This research has confirmed that changing rostering conditions can have a major impact on the efficiency of the constructed rosters. Moreover, this research has demonstrated that this impact can be quantified and can therefore be considered a stepping stone for future research on the impact of crew rostering conditions.

The applicability of the developed simulation model has been tested through two scenario analysis experiments. However, the possible applications of a scenario analysis tool reach way beyond these two initial experiments. Besides assessing the impact of changing rostering policies, the simulation model could be used to evaluate changes in crew composition, flight schedule changes and also the impact of pre-assigned activities. For example, the simulation model could be used for identifying bottlenecks in tight and difficult rosters through constructing, evaluating and comparing rosters with different pre-assigned activities rescheduled or canceled.

There are also a couple of considerations regarding the applicability of the model that should be noted. Firstly, the simulation model takes historical rosters as input. The simulation model can be adjusted such that it is also able to produce rosters for future roster periods. However, this is under the condition that all activities other than pairings and reserve blocks are pre-assigned. If this is not the case, the simulation model should be extended to include the rostering of ancillary activities such as training activities.

Furthermore, the implemented pairing assignment algorithm, the Hungarian Improvement algorithm, is considered to be specifically suitable for the weekly rostering problem as defined in this research. The Hungarian Improvement algorithm exploits the fact that the stated problem resembles a bipartite matching problem. If the roster horizon is extended or if the average length of the pairings decrease, the performance of the Hungarian Improvement algorithm may deteriorate.

Additionally, the Hungarian assignment algorithm used in the Hungarian Improvement algorithm, finds the bipartite solution to a given pilot-pairing cost matrix by performing a set of matrix operations. While constraints concerning a single pilot can be included in such a cost matrix, the Hungarian assignment algorithm cannot cope with constraints that concern multiple pilots.

Three recommendations for future work have been identified. First of all, the developed simulation model can be extended and further improved. For the simulation model to be applicable for a broader variety of scenario analyses, the simulation model can be extended to include the rostering of ancillary activities such as vacations and training activities.

Secondly, future research could focus on improving the efficiency of the solution method implemented in the simulation model, the Hungarian Improvement algorithm. An interesting direction for future research could be to investigate the possibility of applying machine learning techniques to identify which pilot-pairing assignments are expected to improve the remaining bipartite solution and therefore should be evaluated by the Hungarian Improvement algorithm.

Finally, the practical value of the simulation model to airline operators can be increased through the development of a user interface. Especially when the simulation model is extended and includes more and more features, a user interface would facilitate an effective operation of the simulation model.

References

Achour, H., Gamache, M., Soumis, F., & Desaulniers, G. (2007). An Exact Solution Approach for the Preferential Bidding System Problem in the Airline Industry. *Transportation Science*, *41*(3), 354-365. doi: https://doi.org/10.1287/trsc.1060.0172

Bergh, J., van den, Belin, J., de Bruecker, P., Demeulemeester, E., & de Boeck, L. (2013). Personnel scheduling: A literature review. *European Journal of Operations Research*, *226*(3), 367-385. doi: https://doi.org/10.1016/j.ejor.2012.11.029

Blum, C., & Roli, A. (2001). Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Computing Surveys*, *35*, 268-308. doi: https://doi.org/10.1145/937503.937505

Burke, E., & Kendall, G. (2005). *Search Methodologies - Introductory Tutorials in Optimization and Decision Support Techniques*. Springer. doi: https://doi.org/10.1007/0-387-28356-0

Campbell, K., Durfee, R., & Hines, G. (1997). FedEx Generates Bid Lines Using Simulated Annealing. *Interfaces*, *27*(2), 1-16. doi: https://doi.org/10.1287/inte.27.2.1

Cappanera, P., & Gallo, G. (2004). A Multicommodity Flow Approach to the Crew Rostering Problem. *Operations Research*, *52*(4), 583-596. doi: https://doi.org/10.1287/opre.1040.0110

Deveci, M., & Demirel, N. (2018). A survey of the literature on airline crew scheduling. *Engineering Applications of Artificial Intelligence*, *74*, 54-69. doi: https://doi.org/10.1016/j.engappai.2018.05.008

Doi, T., Nishi, T., & Voß, S. (2018). Two-level decomposition-based matheuristic for airline crew rostering problems with fair working time. *European Journal of Operational Research*, *267*(2), 428-438. doi: https://doi.org/10.1016/j.ejor.2017.11.046

Eltoukhy, E., Chan, F., & Chung, S. (2017). Airline schedule planning: a review and future directions. *Industrial Management & Data Systems*, *117*(6), 1201-1243. doi: https://doi.org/10.1108/IMDS-09-2016-0358

Ernst, A., Jiang, H., Krishnamoorthy, M., Owens, B., & Sier, D. (2004). An Annotated Bibliography of Personnel Scheduling and Rostering. *Annals of Operations Research*, *127*(1-4), 21-144. doi: https://doi.org/10.1023/B:ANOR.0000019087.46656.e2

Frank, A. (2005). On kuhn's hungarian methoda tribute from hungary. *Naval Research Logistics (NRL)*, *52*(1), 2-5.

Gamache, M., Soumis, F., & Marquis, G. (1999). A Column Generation Approach for Large-Scale Aircrew Rostering Problems. *Operations Research*, *47*(2), 247-263. doi: https://doi.org/10.1287/opre.47.2.247

Hillier, F., & Lieberman, G. (2015). *Introduction to operations research*. McGraw-Hill Education.

Jarrah, A., & Diamond, J. (1997). The Problem of Generating Crew Bidlines. *Interfaces*, *27*(4), 1-90. doi: https://doi.org/10.1287/inte.27.4.49

Jones, R. (1989). Development of an Automated Airline Crew Bid Generation System. *Interfaces*, *19*(4), 44-51. doi: https://doi.org/10.1287/inte.19.4.44

Kasirzadeh, A., Saddoune, M., & Soumis, F. (2017). Airline crew scheduling: models, algorithms, and data sets. *EURO Journal on Transportation and Logistics*, *6*(2), 111-137. doi: https://doi.org/10.1007/s13676-015-0080-x

Kohl, N., & Karisch, S. (2004). Airline Crew Rostering: Problem Types, Modeling, and Optimization. *Annals of Operations Research*, *127*(1-4), 223-257. doi: https://doi.org/10.1023/B:ANOR.0000019091.54417.ca

Lucic, P., & Teodorovic, D. (2007). Metaheuristic approach to the aircrew rostering problem. *Annals of Operations Research*, *155*(1), 311-338. doi: https://doi.org/10.1007/s10479-007-0216-y

Maenhout, B., & Vanhoucke, M. (2010). A hybrid scatter search heuristic for personalized crew rostering in the airline industry. *European Journal of Operational Reserach*, *206*(1), 155-167. doi: https://doi.org/10.1016/j.ejor.2010.01.040

Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, *5*(1), 32–38.

Ryan, D. (1992). The solution of massive generalized set partitioning problems in aircrew rostering. *Journal of the Operational Research Society*, *43*(5), 459467. doi: https://doi.org/10.1057/jors.1992.72

Part II

Literature Study

Previously graded under AE4020

Executive Summary

After an airline has determined which flights to fly and what aircraft to operate, the final stage in airline schedule planning is to decide which crew members should be assigned to the scheduled flights. This problem is called the 'crew scheduling problem'. Due to the sheer size and complexity of the crew scheduling problem, airlines usually divide the problem into two phases: the 'crew pairing problem' and the 'crew rostering problem'. The crew pairing problem is aimed to construct efficient sequences of flights, i.e. 'pairings', that can be flown by a single crew member. Subsequently, in the crew rostering phase crew members get assigned to these pairings and by doing so a roster for each individual crew member is constructed. This latter phase, the airline crew rostering problem, is the focus of this literature study.

This literature study provides an overview of the academic literature on airline crew rostering. The state-of-the-art is discussed for each of the three general approaches to the airline crew rostering problem: the *bidlines approach*, the *equitability-based personalized rostering approach* and the *Preferential Bidding System*. It has been found that these approaches have been studied mostly independently in academic literature. Academic literature on airline crew rostering has been focused on developing more efficient solution methods within the framework of a certain crew rostering approach and the associated rules and regulations. Future studies should have a broader view on the airline crew rostering problem and could assess the impact of various aspects of a crew rostering approach, i.e. *crew rostering elements*, on the efficiency and quality of the generated rosters. In this regard, the following research gaps have been identified:

1. **Integration of the rostering process -** the assignment of pairings, vacations, training activities and reserve blocks could be integrated into one rostering process. It can be investigated whether the integration of the rostering process of these different activities results in increased roster efficiency;

2. **Impact of equitability and seniority measures -** equitability and seniority considerations can be taken into account as constraints or objective function coefficients in the crew rostering model. The impact of these different measures on the cost and the quality of the generated roster can be evaluated;

3. **Simulation of the dynamic rostering process -** the rolling horzion procedure can be simulated. With a dynamic roster model, global constraint consistency, the influx of crew requests and the effect of disturbances can be investigated;

4. **Consistent evaluation of crew rosters -** a method for the consistent evaluation of rosters can be developed. This method should take the generated rosters as input and should be independent of the approach and method adopted in the generation of the rosters.

The identified research gaps require the simulation and testing of different crew rostering approaches. This literature study provides an overview of the modelling techniques that could be used to develop an airline crew rostering model. Especially for the research purposes identified in this literature study, the adaptivity and computational efficiency of the crew rostering model are essential. The research objective therefore is the following:

Research objective: *to provide insight into the impact of crew rostering elements on the efficiency and quality of the generated rosters, by developing a fast and adaptive crew rostering model, simulating different rostering approaches and evaluating the generated crew rosters.*

The assessment of the impact of crew rostering elements can provide airlines with insights on how the efficiency of their adopted rostering approach could be improved or how they can improve the quality-of-life aspects of their crew schedules. Moreover, a fast and adaptive crew rostering model could serve as a simulation tool for airlines during Collective Labour Agreement (CLA) negotiations with labour unions. Such a simulation model can quantify the impact of new rules and regulations on the efficiency and quality of the crew rostering process. This is considered a major asset for airlines during CLA negotiations, assisting airlines in efficiently picking their battles.

Definitions

Bidlines: an airline crew rostering approach where anonymous monthly rosters are constructed and assigned to crew members based on crew bids.

Crew rostering approach: an approach to the crew rostering problem itself, i.e. a specific problem formulation.

Crew rostering elements: the aspects of a crew rostering model that constitute a crew rostering approach.

Crew rostering method: an approach to solving the crew rostering problem, i.e. a specific solution technique.

Equitability-based personalized rostering: a personalized crew rostering approach where equitability among crew members is pursued, for example in the number of flying hours or in the satisfaction of crew preferences.

Equitability: even balance in roster characteristics among crew members, i.e. similar levels of workload and number of assigned preferred activities.

Equitability measure: a method to ensure equitability among crew members in a crew rostering model.

Personalized rostering: a crew rostering approach that considers pre-assigned personal activities of crew members.

Preferential Bidding System: a personalized crew rostering approach where the satisfaction of the preferences of senior crew members is prioritized over those of junior crew members.

Roster horizon: the number of days or weeks before operation for which a crew roster is generated.

Roster period: the number of days or weeks of the period that is rostered.

Seniority measure: a method to prioritize the satisfaction of senior crew members over the satisfaction of junior crew members in a crew rostering model.

Strict seniority: a way to differentiate in crew preference satisfaction by maximizing crew preference satisfaction sequentially in order of seniority.

Weighted bids: a way to differentiate in crew preference satisfaction by assigning seniority-based weight factors to the crew satisfaction components in the objective function.

1. Introduction

The airline industry is confronted with some of the largest scheduling problems of any industry and the size and complexity of these problems are only growing (Gopalakrishnan and Johnson, 2005). One of these scheduling problems of particular interest is the crew scheduling problem because of its high associated costs. Crew costs is the second largest operating cost component of an airline, only after the expenses for fuel (Barnhart and Cohn, 2004; Belobaba et al., 2009). Even a small increase in the efficiency of the crew scheduling process can therefore induce significant cost savings (Kasirzadeh et al., 2017).

Crew rostering is a sub-problem of the crew scheduling problem and involves the assignment of pairings, i.e. efficient sequences of flights, to individual crew members (Eltoukhy et al., 2017). The main objective is to 'cover' all pairings, i.e. to make sure that every pairing has a crew member assigned. Leaving pairings 'open' induces high costs for airlines since these open pairings need to be resolved with expensive overtime payment, the hiring of freelancers or by cancelling the flight (Kohl and Karisch, 2004). Other objectives in the rostering process include constructing rosters with desirable working patterns, satisfying crew requests for activities or days-off and ensuring equitability among crew members (Kohl and Karisch, 2004). These objectives are pursued within the boundaries of governmental and labour union regulations. With increasing number of flights that need to be assigned and the accumulation of labour union regulations, efficiently solving the crew rostering problem has become a major challenge for airlines (Eltoukhy et al., 2017; Maenhout and Vanhoucke, 2010).

The purpose of this literature study is threefold: to provide an overview of the academic literature on airline crew rostering, to identify potential subjects for future studies and to identify modelling techniques that can be used to address the identified research gaps.

The literature study is structured as follows. First, the academic literature on the airline crew rostering problem is discussed. Afterwards, in Chapter 3 the identified research gaps are introduced and explained. Then, in Chapter 4 a broader view on the personnel rostering problem is provided by investigating rostering approaches in other industries. Chapter 5 is focused on providing background information on methods that can be used in addressing the research gaps. Afterwards, Chapter 6 contains a concise overview of the literature that is considered in this literature study. Finally, some of the main insights obtained in this literature study are discussed in the conclusion.

2. *Scope:* Airline Crew Rostering Approaches

Crew rostering can be done in various ways following different 'approaches'. In the airline industry, a distinction is made between three general crew rostering approaches (Deveci and Demirel, 2018; Kasirzadeh et al., 2017):

- the *bidlines approach*;
- the *equitability-based personalized rostering approach*;
- the *Preferential Bidding System (PBS)*.

Outside of North America, equitability-based personalized rostering is the most common approach for crew rostering (Kasirzadeh et al., 2017). The bidlines approach is the approach that used to be common practice in North American airlines. However, nowadays some large North American airlines have adopted the Preferential Bidding System (PBS) as their crew rostering approach (Achour et al., 2007; Kasirzadeh et al., 2017).

The state-of-the-art in crew rostering for each of the three different rostering approaches is discussed separately in the following sections. In this literature study, in order to differentiate between how the crew rostering problem is formulated and how the crew rostering problem is solved, the following definitions are adopted:

Definition: A *crew rostering approach* is defined as an approach to the crew rostering problem itself, i.e. a specific problem formulation.

Definition: A *crew rostering method* is defined as an approach to solving the crew rostering problem, i.e. a specific solution technique.

2.1 Bidlines Approach

In the *bidlines approach*, a set of anonymous monthly rosters (*bidlines*) are constructed, where after crew members place bids on the rosters they prefer (Jarrah and Diamond, 1997). Subsequently, the rosters are assigned to specific crew members based on seniority, meaning that if two or more crew members have placed a bid on the same roster the roster will be assigned to the most senior crew member (Kohl and Karisch, 2004). After assignment of the bidlines, numerous pairings conflict with the individual pre-assigned activities of crew members such as training activities or granted days-off. Those pairings that are in conflict with pre-assigned activities are then removed from these bidlines. The least senior crew members will then be assigned to a combination of 'open pairings', i.e. pairings not covered by any of the generated bidlines (including those due to conflicts with pre-assigned activities), and reserve days (Campbell et al., 1997).

Definition: fhe *bidlines approach* is defined as an airline crew rostering approach where anonymous monthly rosters are constructed and assigned to crew members based on crew bids.

An early successful application of operations research (OR) techniques for the bidline assignment problem is a case study by Marsten et al. (1979) at Flying Tiger Line, a former United States cargo airline. Larger airlines such as United, Air France and Air Canada had already tried to implement automated scheduling, but with the computers and OR techniques then available their fleets were found to be too large and the results were disappointing. Marsten et al. (1979) solve the pairing selection and bidline assignment problems for a roster horizon of one week, because the monthly problem is deemed to be intractable. The outcomes of four consecutive weekly problems are then put together to form a set of monthly bidlines. The bidline selection program exploits a branch-and-bound algorithm and use the dual simplex method. The bidline selection process is based on two bidline cost components, as also reported in a later paper on the same case study by Marsten and Shepardson (1981): the variance from an ideal number of flying hours and the closeness to 08:00 AM arrivals (quoting: *"8:00 AM arrivals are undesirable because days off must start at 8:OO AM so a delay could mean a lost day"*). The model of Marsten and Shepardson (1981) was limited to constructing a monthly bidline by combining four identical weekly rosters. Therefore, the model could not solve problems for flight schedules that vary week by week.

Ten years later, Jones (1989) describes the development of an automated bidline generation system at a major airline with a much bigger fleet, flight schedule and crew pool: American Airlines. The bidline generation system of Jones (1989) consist of two heuristic processes. First, the *DEAL* section generates an initial set of bidlines directly from the pairings with the constraint on monthly flying time relaxed. Then, the *SWAP* section takes these initial set of bidlines as input and tries to exploit moving and swapping opportunities to adjust the bidlines such that the flying time for each bidline is within the minimum and maximum allowable limits. The results of the described bidline generation system show that it was capable of producing 60 bidlines per hour, in contrast to a manual construction rate of 10 bidlines per hour, therefore hugely increasing the efficiency of the crew scheduling process of American Airlines (Jones, 1989).

Incorporation of quality parameters

Another important step in solving the bidline generation problem was made by Jarrah and Diamond (1997). Jarrah and Diamond (1997) are the first who incorporate 'quality parameters' in the bidline generation process. Where previous bidline generation systems only aim for the minimization of crew costs, the model of Jarrah and Diamond (1997) also takes into account the quality of the bidlines from the crew's perspective. For example, bidlines with long stretches of off-days in between blocks of pairings are more desirable than bidlines with the pairings uniformly dispersed over the roster. These '*quality-of-life*' parameters are modeled as constraints in the bidline generation model, and can be adjusted to the airline's specific demands. The bidline generation problem is then modeled as a set-partitioning problem with the objective function to maximize the covered flying time while using the least possible bidlines. The model then uses column generation to generate new bidlines as long as the objective function improves with every bidline added to the Restricted Master Problem (RMP) (Hillier and Lieberman, 2015; Jarrah and Diamond, 1997). The column generation is governed by a greedy heuristic and an improvement heuristic. The greedy heuristic prioritizes the rostering of 'open pairings', i.e. pairings still not covered by any of the generated bidlines, in the next bildine generation iteration. The improvement heuristic on the other hand prioritizes the rostering of pairings that appeared in earlier generated 'good bidlines' in terms of their objective function value. The results of the bidline generation system show enormous benefit in terms of scheduling productivity: the most complex problem in their case study at a major US airline (1649 pairings and 223 bidlines) was solved in just 11 minutes, whereas it took 15 man-days to solve the problem manually (Jarrah and Diamond, 1997). The results also show that too 'greedy' quality-of-life parameter settings can result in a problem that is over-constrained, in one of their case studies resulting in an unacceptable percentage of uncovered flying time (10.82 percent). However, after readjusting the parameters the program produced satisfactory results. This shows that the usability of the model relied on the capability of the scheduler to manually tune the quality-of-life parameters, which is considered a limitation of the model of Jarrah and Diamond (1997).

Introduction of meta-heuristics

At the end of the 1990's the state-of-the-art in bidline generation involved the use of meta-heuristics. Campbell et al. (1997) propose a solution methodology for the bidline generation problem based on simulated annealing. The simulated annealing algorithm is used to generate as many valid bidlines as possible. Afterwards, a greedy heuristic takes the remaining open pairings and generates additional bidlines with these remaining pairings. The objective function of the model of Campbell et al. (1997) contains three elements: minimization of open time, minimization of the number of bidlines and the maximization of bidline '*purity*'. Bidline purity is a measure of how well certain roster quality-of-life considerations are accommodated in the bidlines. One of the identified limitations of the model is the transition between the simulated annealing algorithm and the greedy algorithm. The model of Campbell et al. (1997) runs the simulated annealing process for a set number of maximum steps. After these set number of steps, the program continues with the greedy algorithm that produces additional bidlines based on the remaining uncovered pairings.

Christou et al. (1999) developed a bidline generation model based on another meta-heuristic: the genetic algorithm. This model has improved computational efficiency over the model of Campbell et al. (1997) because the model of Christou et al. (1999) exploits a-priori knowledge about what constitutes a 'good' bidline before the meta-heuristic bidline generation technique is initiated. The model is partitioned into two phases: the 'purity phase' and the 'GA phase'. During the purity phase, high-quality bidline patterns are generated with this a-priori knowledge. For example, bidlines with pairings departing on the same day of the week in multiple weeks of the bidline is said to be '*day-pure*'. Similarly, a bidline is called '*trip-pure*' if it contains multiples of the same pairings, departing on different days of the roster. After the purity phase, the model contains a set of incomplete bidlines with high quality pairing patterns. These incomplete bidlines are then completed in the GA-phase through the use of a genetic algorithm. The genetic algorithm is considered particularly useful in this regard, because of its capability to quickly locate promising search regions (Christou et al., 1999). The model developed by Christou et al. (1999) is tested and implemented at Delta Air Lines. The results show slight improvements in bidline costs in terms of covered flying time compared to the semiautomatic approach adopted originally by Delta. However, the main benefits of the proposed model are found to be in the increased computational efficiency and the construction of bidlines with higher 'purity', i.e. better quality-of-life characteristics.

Weir and Johnson (2004) partition the bidline generation problem into three separate phases. The first phase aims to develop efficient patterns of pairings. The second phase then builds bidlines from these generated patterns. The third phase then constructs additional bidlines to account for pairings still not covered after phase 2. In contrast to previous research, Weir and Johnson (2004) adopt classical Mixed Integer Programming (MIP) techniques in the development of their bidline generation model. The MIP's in the three phases are relaxed to an LP and solved using column generation, branch-and-bound and interior point method techniques. Weir and Johnson (2004) test their model for problems with up to 150 bidlines to be generated. The longest run time reported was 2 hours and 16 minutes. However, the model is significantly less efficient in terms of computational time compared to the meta-heuristic techniques of Campbell et al. (1997) and Christou et al. (1999).

Incorporation of equitability

Whereas previous research focused on producing efficient and high-quality bidlines, Boubaker et al. (2010) incorporate equitability into the bidline generation problem. The model of Boubaker et al. (2010) minimizes the variances in quality or purity between the generated bidlines. In order to do so, the objective function is to minimize the weighed sum of the squared deviations of the number of flying hours and number of days off from the mean. The number of available crew members are considered a fixed input parameter rather than that the model minimizes the number of crew members required to cover all pairings. Boubaker et al. (2010) model the bidline problem with equitability as a set-partitioning problem. First, a standard branch-and-price heuristic based on a network representation of the problem was used to solve the problem. Afterwards, the branch-and-price heuristic has been enhanced by adding the dynamic constraint aggregation heuristic as introduced by Elhallaoui et al. (2005). The model is tested on eight problem instances, both with just the branch-and-price heuristic and with the branch-and-price heuristic enhanced by the dynamic constraint heuristic. The addition of the dynamic constraint aggregation heuristic shows spectacular results. First of all, the computational time reduces by a factor of 30 and remains within an hour for all problem instances including the largest with 2924 pairings and 564 crew members. Furthermore, the enhancement with the dynamic constraint aggregation heuristics also yields significantly better results, reducing the variances in flying hours and days off by 50-60% on average. Based on the results discussed above, it can be stated that the solution methodology of Boubaker et al. (2010) shows huge potential for solving these kinds of massive bidline generation problems.

Two years later, Saddoune et al. (2012) have use this same dynamic constraint aggregation heuristic to solve the integrated airline scheduling problem in the bidline context. The integrated bidline crew scheduling problem integrates the crew pairing and the bidline generation problem into one complex combinatorial problem. This highly complex problem requires a lot of computational power and highly efficient solution algorithms (Saddoune et al., 2012). However, the results of Saddoune et al. (2012) have shown that the integrated approach outperforms the sequential approach in terms of cost savings. On average, the integrated bidline crew scheduling problem requires 5.54% fewer crew members and yields a 3.37% reduction in total costs. However, the integrated bidline scheduling problem takes on average 6.8 times longer to solve compared to solving the crew pairing and bidline generation problem sequentially. Consequently, the biggest tested problem instance involving 7527 flights to be scheduled took the integrated model more than 44 hours to solve. The need for the development of more efficient algorithms for the integrated bidline crew scheduling problem is therefore acknowledged by Saddoune et al. (2012).

Synthesis

Academic literature on the bidline generation problem focuses on three research efforts: making the bidline generation computationally more efficient such that larger problems can be solved in a shorter amount of time, incorporating quality-of-life considerations within individual bidlines and ensuring an equitable qualtiy-of-life distribution among the generated bidlines. However, what is not considered in literature, is the process after the bidlines are assigned to crew members and pairings are found to be conflicting with the pre-assigned activities of these crew members. Campbell et al. (1997) states that these open pairings resulting from these conflicts will be allocated to the least junior members along with a number of reserve blocks. However, this additional problem is not addressed in literature.

2.2 Equitability-based Personalized Rostering

With *personalized rostering approaches* personalized rosters are constructed for each individual crew member, considering pre-assigned personal activities such as training days and vacations (Kasirzadeh et al., 2017). The more advanced personalized rostering systems also take into account individual crew preferences on certain activities or roster attributes in the roster optimization (Kohl and Karisch, 2004). Two types of personalized rostering can be distinguished: the *equitability-based personalized rostering approach* and the *seniority-based personalized rostering approach*, also called the *Preferential Bidding System (PBS)*. Equitability-based personalized rostering involves constructing equitable rosters for all crew members whereas in a Preferential Bidding System the satisfaction of the preferences of senior crew members is prioritized over the satisfaction of those of junior crew members. This subsection discusses the equitability-based personalized rostering approach. The Preferential Bidding System is the subject of Section 2.3. The following definitions are adopted:

Definition: a *personalized rostering approach* is defined as a rostering approach that considers the pre-assigned activities of crew members.

Defintion: *equitability* is defined as an even balance in roster characteristics among crew members, i.e. similar levels of workload and satisfaction of crew preferences.

Definition: An *equitability-based personalized rostering* approach is defined as a personalized crew rostering approach where equitability among crew members is pursued.

Nicoletti (1975) introduced operations research (OR) techniques in the literature on equitability-based personalized crew rostering. Nicoletti (1975) stated that due to the high complexity of the airline crew rostering problem, the problem needed to be separated into pieces and solved using heuristics. Two approaches to do so are considered, being either crew member-by-crew member or day-by-day roster construction. Nicoletti (1975) argued that equitability among crew members can be better achieved by approaching the rostering problem day-by-day, which is why that approach is used in his model. The model is formulated on a network and it is solved using the '*out-of-kilter*' algorithm to compute the minimum cost flow in the network (Ford and Fulkerson, 1962). The model was tested on a small test case involving 103 crew members and rosters were assessed on their equitability in terms of the balance in number of flight hours. Although results are reported ambiguously, it can be seen that the computerized model was found to outperform manual roster construction in terms of equitability.

Exact solution techniques

Where the previously discussed model used a sequential day-by-day heuristic to solve the rostering problem, Ryan (1992) proposed an exact mathematical optimization technique to solve the personalized rostering problem by modelling it as a generalized set-partitioning problem. In the model of Ryan (1992) first for each crew member all possible legal monthly rosters are generated, taking pre-assigned other activities into account. Two filtering techniques are used to speed up this process and to generate overall better rosters. Afterwards, for each crew member one of his or her generated rosters need to be selected, which is the set-partitioning problem. The objective function used for the selection of the rosters contains both the minimization of crew members and the maximization of equitability. Although the model of Ryan (1992) itself does not contain crew preferences, Ryan (1992) does discuss two ways to include crew preferences into the model. First, it can be included in the generation of all possible legal rosters by influencing the filters and secondly, they can be included in the objective function during the selection of the rosters. The LP-relaxation of the set-partitioning problem is solved with integer programming techniques, using the simplex method and a branch-and-bound procedure. The model is tested on numerous problem instances ranging from around 50 to 475 pilots and from 100 to 200 pairings. The largest problem, with 461 pilots and 200 pairings can still be solved in under two hours. Also, Ryan (1992) states that the advantage of mathematical optimization techniques over heuristics is that infeasibility can be identified with certainty and can also pinpoint which days or trips cause the infeasibility. This is crucial information in an operational crew scheduling environment. For example, training activities on tight schedule days can be rescheduled to "free" additional crew resources on these days.

A subsequent influential contribution to the literature on equitability-based personalized rostering was made by Gamache et al. (1999). Gamache et al. (1999) also formulate the rostering problem as a set-partitioning problem and solves it also using a branch-and-bound procedure. However, whereas the branch-an-bound procedure of Ryan (1992) is based on exhaustive enumeration with some filtering techniques, Gamache et al. (1999) proposes a column generation scheme for the branch-and-bound procedure. With the column generation scheme, the master problem is the generalized set-partitioning problem and the subproblems (one for each crew member) are represented on a network. The network of each crew member is personalized, containing crew-specific arcs for pre-assigned activities. The objective function consists of maximizing the duration of the pairings covered by regular crew members. As such it is equivalent to minimizing the duration of open pairings. The LP-relaxation of the set-partitioning master problem is solved using the simplex method and the dual variables are used for the pricing problem in the column generation scheme. For each crew member there is a constrained shortest-path subproblem and these subproblems add columns to the master problem (Gamache et al., 1999). The shortest-path subproblem is solved using a dynamic programming procedure based on Desrosiers et al. (1995). The model showed to be capable of solving very large-scale problems in limited computational time. One problem instance, involving 386 subproblems (crew members) was solved in 45 minutes.

Fahle et al. (2002) also follow the "generate-and-optimize" principle, but solve the column generation subproblem using Constraint Programming (CP) techniques. The CP algorithm is used to solve the pricing problem in the column generation scheme. A shortest path constraint and a negative reduced cost constraint are programmed dynamically, meaning that in every iteration the size of the problem is reduced using updated dual information of the master problem. The CP algorithm is used in order to identify illegal choices of the shortest-path algorithm as early as possible, which increases the computational efficiency of the column generation algorithm. The results show that the computational time of the column generation scheme enhanced with CP techniques can be reduced significantly compared to the traditional column generation approach.

Crew rostering in an operational context

Cappanera and Gallo (2004) formulate the rostering problem as a 0-1 multicommodity flow problem, where each crew member corresponds to a commodity. The monthly roster of a crewmember can then be seen as a path over the graph where each arc represents either an activity (pairings, training etc.) or a compatibility (travel leave, day-off etc.). As such, the objective is to minimize the number of uncovered pairings. Cappanera and Gallo (2004) also acknowledge equitability as an important objective during the rostering process. However, Cappanera and Gallo (2004) state that because equitability concerns are more and more integrated in collective labour agreements, these equitability concerns are treated as constraints rather than an additional objective in the modelling of the problem. Therefore, the objective function is solely focused on minimizing costs. A remarkable feature of the model of Cappanera and Gallo (2004) is the fact that it considers the rostering problem in the operational context rather than seeing it as a static problem solved at one instance in time. The model of Cappanera and Gallo (2004) takes the previous roster period into account, by fixing for each commodity (crew member) the beginning of its path along the activity arcs that resulted from the previous roster period. The model is tested using two variations of the objective function: the *max cardinality* case consists of maximizing the number of covered activities and the *max length* case consists of maximizing the total length of covered activities. The model is tested for both cases on small-scale problems of about 50 crew members using the IBM CPLEX software. The results show that the max cardinality case outperforms the max length case both in terms of roster quality and computational time. In terms of computational time, the model of Cappanera and Gallo (2004) shows remarkable results. Although the different test problems have similar sizes of around 50 crew members and 270 pairings, the time required for the model to solve the problems varies significantly, ranging from minutes to multiple hours.

Kohl and Karisch (2004) provide an overview of the personalized rostering problem and present a model that is used in the commercial *Carmen* software, which is employed in various airline such as British Airways, Alitalia and KLM. The model of Kohl and Karisch (2004) is based on the earlier discussed 'generate-and-optimize' principle. First, a construction method is used to construct an initial feasible roster solution. The construction method is based on an iteratively applied 1-matching in a bipartite graph. In every iteration, one activity is assigned to one crew member sequentially and the individual rosters are updated for the next iteration. This phase stops when either all activities are assigned or no more activities can be legally assigned. After this initial solution has been constructed, column generation is used to generate rosters that would improve the overall solution. Equitability of rosters is also discussed in the paper of Kohl and Karisch (2004). One of the insights provided by Kohl and Karisch (2004) is the fact that equitability within one roster period is not the main issue, but it is equitability over the whole year what airlines and crew members are most interested in. This notion would provide more flexibility to the roster model, but would also require additional complex rules that take previous rosters into account during the rostering process. Moreover, Kohl and Karisch (2004) notices that equitability is a nonlinear function, since a few large deviations are deemed to be worse than many small deviations. The Carmen system is capable of solving huge problems, the largest problem instance with 1600 crew members and 5000 activities to be assigned was solved by the Carmen system in around 10 hours. Another insight provided by Kohl and Karisch (2004) is that besides problem size the complexity of the rules and objectives also have a major impact on solution run times.

Introduction of meta-heuristics

Just as with the bidline problem, around the year 2000 meta-heuristics were introduced to the field of personalized crew rostering. Lucic and Teodorovic (2007) test three different meta-heuristics as solution techniques for the personalized rostering problem: simulated annealing, the genetic algorithm and tabu search. Simulated annealing and tabu search are so-called *local search* techniques, meaning that they are focused on the solution space in the neighborhood of the current solution. The genetic algorithm on the other hand is based on *population search*, evaluating whole groups of solutions simultaneously and then attempting to generate a new improved generation of solutions. For all three meta-heuristic models first an initial solution needs to be provided. Lucic and Teodorovic (2007) exploit a standard pilot-by-pilot greedy heuristic to generate an initial feasible solution. Thereafter, the meta-heuristic is used to improve the current solution. The objective function consists of three different elements such that equitability is pursued on three roster attributes: the deviations from the ideal in flight time, duty days spent in foreign areas and number of weekend days on duty are all minimized. The three different models are tested on five problem instances, ranging from 27 to 65 pilots and 221 to 580 pairings to be rostered. A remarkable insight provided by the results is that all three models provide better results for larger problem instances. This can be explained by the fact that all three objective function elements are equitability-based. In larger problems there is more flexibility in the roster to establish equitable rosters, resulting in 'better' rosters measured in terms of equitability. Lucic and Teodorovic (2007) state that the best results were obtained by the simulated annealing meta-heuristic. However, the results also show that the simulated annealing algorithm requires more computational time to solve relative to the other meta-heuristics.

Maenhout and Vanhoucke (2010) propose the use of another meta-heuristic to solve the personalized rostering problem: the scatter search meta-heuristic. Scatter search is a population-based meta-heuristic and as such similar to the genetic algorithm. The objective function adopted by Maenhout and Vanhoucke (2010) consists of three components: minimization of costs in terms of duration of open pairings, the maximization of equitability by penalizing deviations from certain standard or ideal roster attribute values and the maximization of preference satisfaction of employees. The model is formulated as a set-partitioning problem. First, a constructive heuristic is used to generate an initial solution and afterwards, the scatter search meta-heuristic tries to improve this initial solution. The results of their new solution methodology are compared to the exact branch-and-price procedure of Gamache et al. (1999) and to a variable neighborhood search procedure based on Hansen and Mladenovic (2001). The different models were tested on 50 generated artificial problems based on a real-life problem instance provided by Brussels Airlines. The problem instances involved 100 crew members and 700 activities and rosters were needed to be created for a 2-week period. Vacation days, pre-assigned activities and crew preferences were randomly generated for the crew members. The hybrid scatter search heuristic shows promising results. In an average computational time of only 127 seconds the hybrid scatter search heuristic obtains results that deviate an average of 1.30% from the result obtained by the branch-and-price algorithm with an average run time of over 20 hours. In 60% of the problem instances, the branch-and-price algorithm solves to optimality, and the deviation in terms of solution quality of the hybrid scatter search heuristic then amount to only 1.67%. Also, the hybrid scatter search heuristic outperforms the variable neighborhood search procedure by 3.20% at the cost of just a couple of seconds of average computational time. Because of its capability of obtaining near-optimal results in just two minutes of run time, the model of Maenhout and Vanhoucke (2010) was used as a simulation tool in the negotiations between Brussels Airlines and the pilot union. This simulation tool was considered to be very valuable in the process of determining the appropriate configuration of rostering rules such as the ratio of working days versus idle days and the minimum rest between pairings.

Recent contributions

Kasirzadeh et al. (2017) use the traditional solution methodology based on a branch-and-bound procedure using column generation to solve the monthly personalized rostering problem. The pricing problem in the column generation scheme is solved using a multi-label shortest path algorithm taken from Desrosiers et al. (1995). Two types of preferences are included in the objective function, being preferred air legs and preferred vacations. Both these preferences are included in the cost function by penalizing for preferences that are not satisfied. The cost parameters for the penalization of unsatisfied preferences are determined based on trial and error, tuning the costs and checking the preference satisfaction results. As a baseline, the cost of covering a scheduled flight is 0. The penalty cost of not covering a scheduled flight is then set to 10,000. The penalty cost of failing to satisfy a vacation preference is then set to 1000 and a negative cost of -100 is given to any flight preference that is satisfied. In addition, Kasirzadeh et al. (2017) include preference satisfaction constraints in order to ensure the satisfaction of a minimum number of preferred vacations and preferred flights. The model of Kasirzadeh et al. (2017) can be considered to be a good reference for constructing a basic rostering model. However, the model could be improved by implementing a data-driven process to compute the cost parameters in the objective function.

Armas et al. (2017) develop a multi-start randomized heuristic that was aimed at providing multiple feasible solutions in short computational time. Beneficial characteristics of the multi-start randomized heuristics are considered their flexibility, computational efficiency and simplicity. The airline for which the model was developed used commercial software during the rostering process, but a number of constraints were not implemented in the software resulting in hours of manual post-processing before the rosters could be assigned to crew members. The model of Armas et al. (2017) was therefore tasked to avoid this need for manual adjustment, by incorporating these additional constraints into one roster model. The model of Armas et al. (2017) has resulted in significant improvements over the commercial software of the airline. The model provided better results in under 1 minute than the rosters that were produced using the commercial software and manual post-processing. However, the problem sizes of the tested problem instances are very small, ranging to a maximum of 20 crew members and 17 pairings to be scheduled in a monthly roster. Two somewhat larger problem instances of around 40 crew members and around 20 pairings are also considered, but the solution quality of the multi-start randomized heuristic is not compared to any benchmark. As such, the test cases are not a realistic reflection of the crew rostering problems encountered in larger airlines.

The latest contribution to equitability-based personalized crew rostering literature considered in this literature review is a study by Doi et al. (2018). Their model is exploits a '*matheuristic*', which is a combination of mathematical programming techniques and a meta-heuristic. The matheuristic used in the model is called 'POPMUSIC', which is an acronym for 'Partial Optimization Metaheuristic Under Special Intensification Conditions'. The POPMUSIC matheuristic tries to improve the initial solution by locally optimizing subparts of the solution. Just like other meta-heuristics, the POPMUSIC matheuristic is an improvement technique and thus requires an initial solution as input. Similar to previous research efforts, Doi et al. (2018) use a pilot-by-pilot greedy heuristic to develop an initial roster solution. The objective function used for improving this initial solution is aimed at pursuing equitability by minimizing the deviations per crew member from the standard working time. Although the model thus is aimed at equitability, the model of Doi et al. (2018) also considers seniority in their model. The complete crew pool is divided into groups according to crew seniority and qualifications, since in their model the maximum total working time of junior crew members is greater than that for a senior crew member. Three types of objective function penalties were tested: linear penalty, quadratic penalty and min-max penalty. These types of penalties refer to how the objective function considers and aggregates the separate deviations from the standard of all crew members. The study has shown that a linear penalty was the most effective measure to pursue equitability in working time. The model has been compared to the mathematical approach of IBM CPLEX and to a Constraint Programming (CP) solver. Tests were performed on 7 problem instances. The matheuristic model of Doi et al. (2018) has shown to outperform both CPLEX and CP, both in solution quality and computational time. In turn, CPLEX is found to be superior to CP for all problem instances, also both in terms of solution quality and computational time.

Synthesis

The personalized rostering problem is considered to be much more difficult to solve than the bidlines problem, because of its increased combinatorial complexity and additional constraints (Grosche, 2009; Ryan, 1992). The majority of the research focuses on developing efficient solution methodologies to solve problems of realistic problem sizes in reasonable computational time. The management of crew preferences is another focus in recent research efforts. Crew preferences can be either included in the pre-assignment of activites, as soft constraints in the objective function or as a hard constraint, for example to ensure that a minimum number of preferences are satisfied.

2.3 Preferential Bidding System

The third airline crew rostering approach is also a personalized rostering approach, but can be distinguished from the equitability-based personalized rostering approach by the fact that it is seniority-based (Kasirzadeh et al., 2017). In a Preferential Bidding System (PBS) the satisfaction of preferences of senior crew members is prioritized over the satisfaction of the preferences of junior crew members. In literature, two ways to do so are considered: *weighted bids* and *strict seniority*.

Definition: a *personalized rostering approach* is defined as a rostering approach that considers the pre-assigned activities of crew members.

Definition: a *Preferential Bidding System (PBS)* is defined as a personalized rostering approach where the satisfaction of the preferences of senior crew members is prioritized over those of junior crew members.

Definition: *weighted bids* is defined as a way to differentiate in crew preference satisfaction by assigning seniority-based weight factors to the crew satisfaction components in the objective function.

Definition: *strict seniority* is defined as a way to differentiate in crew preference satisfaction by maximizing crew preference satisfaction sequentially in order of seniority.

PBS with weighted bids

The introduction of the Preferential Bidding System is described in a paper of Gamache and Soumis (1998). The model as proposed by Gamache and Soumis (1998) addresses the personalized rostering problem with preferences, referred to as *desiderata* in the paper. What differentiates the model of Gamache and Soumis (1998) from equitability-based models, is that the incentive in the model for granting preferences to crew members varies with the priority ranking of the considered crew member. As such, crew members with higher priority have higher chance of getting their preferred pairings or days off. The model is formulated as a generalized set partitioning problem which is solved using column generation and a branch-and-bound procedure. Within the column generation scheme, there is one subproblem for every crew member which is represented on a personalized network. The priority ranking function then is implemented in the cost values of the arcs in each personalized network. The model is tested on small problem instances of 10 to 20 crew members and 70 to 100 pairings to be rostered over a two-week period. These problems are solved in approximately one hour. This is considered quite a long computational time for such a limited problem, compared to the equitability-based personalized rostering approach using a similar solution methodology as described in Gamache et al. (1999). Since these two studies are comparable, being conducted by the same researchers and using similar solution methodologies, it can be concluded that including a priority ranking in personalized rostering further complicates the rostering problem. Gamache and Soumis (1998) compare their approach to crew preference satisfaction with two other approaches: minimum staff size and pre-assignment. The approach of Gamache and Soumis (1998) as described above is considered a compromise between the two other approaches, where in 'minimum staff size' only those crew preferences are satisfied that do not come at any cost and in 'pre-assignment' preferences are granted and pre-assigned as long there remains a feasible solution. Figure 2.1 shows the trade-off between roster efficiency and crew preference satisfaction. The results show that the compromise of including preferences in the rostering model can provide a significant increase in crew satisfaction at just a small cost in terms of roster efficiency.

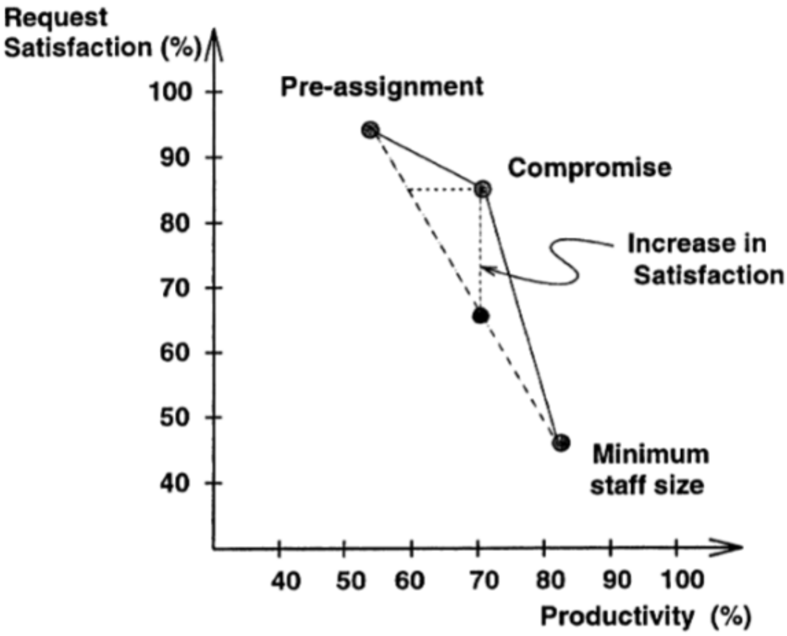

Figure 2.1: The impact of different approaches to crew preference management on roster efficiency and crew preference satisfaction (Gamache and Soumis, 1998)

PBS with strict seniority

In a subsequent paper, Gamache et al. (1998) explicitly refer to their rostering approach as a Preferential Bidding System. The solution methodology is based on sequential pilot-by-pilot assignment based on seniority order. Each subproblem is a residual problem which is solved such that an optimal roster is constructed while also ensuring a feasible solution for the remaining (less senior) crew members. For each residual problem, the objective is to maximize crew preference satisfaction. The approach taken in Gamache et al. (1998) differs from the previous study Gamache and Soumis (1998) in the fact that Gamache et al. (1998) adopt '*strict seniority*' restrictions instead of the '*weighted bids*' adopted by Gamache and Soumis (1998). The strict seniority restrictions govern that a maximum score in terms of preference satisfaction is pursued for every crew member, as long as this maximization is never done at the expense of a more senior crew member. As such, even a minor increase in roster quality of a senior member is prioritized over significantly increasing the quality of multiple junior crew members. The model is tested on 24 problem instances. The problems range from 18 to 108 crew members and 103 to 602 pairings to be rostered over a month. Most smaller problems up to 50 crew members are solved within half an hour. However, larger problems take several hours, with the largest run time reported to be 8 hours. Desaulniers et al. (1998) confirms that the computational times reported in Gamache et al. (1998) are significantly larger than those reported in the personalized rostering problems as discussed in Gamache et al. (1999). Desaulniers et al. (1998) explain this notion by the fact that in a Preferential Bidding System with strict seniority restrictions a set-partitioning problem needs to be solved for each crew member, in order to check whether a feasible solution can still be obtained for the remaining crew members. This significantly increases the complexity of the problem.

Achour et al. (2007) build upon previous research of Gamache et al. (1998) and also considers the PBS with strict seniority restrictions. However, as opposed to the pilot-by-pilot sequential method as used in Gamache et al. (1998), the model of Achour et al. (2007) delays the selection of the roster of a senior crew member in the case where a senior crew member has multiple feasible rosters with a maximum bidding score, i.e. maximum preference satisfaction. So although the model of Achour et al. (2007) also considers crew members sequentially, the rosters are not necessarily fixed in seniority order. The approach of Achour et al. (2007) delays the roster selection process until there is a unique best-score for the considered crew member that yields the best rosters for subsequent junior crew members. The model of Achour et al. (2007) was tested on eight problem instances, involving 17 to 91 crew members, and compared to the model of Gamache et al. (1998). For small problem instances, the model of Achour et al. (2007) took longer to solve but provided better results in terms of overall crew preference satisfaction. Larger problems are also improved in terms of solution quality with the new approach. Moreover, for these larger problems the model of Achour et al. (2007) requires less computational time to solve than the baseline model of Gamache et al. (1998). However, despite these improvements, the computational time of the model is still considered unsatisfactory. Although most problems are solved within 10 minutes, certain problems of the same size give rise to significant issues in the roster model. In these specific problems, the model is confronted with infeasibility and 'backtracking' is evoked, i.e. the model reverses and returns to previous problems in the pilot-by-pilot sequential method. When backtracking occurs, run times are shown to be greatly increased, up to a longest reported run time of 127 hours.

Avoiding backtracking

Where previous research solved the Preferential Bidding System problem using mathematical programming techniques, Gamache et al. (2007) propose a model using graph coloring techniques and the tabu search meta-heuristic. Gamache et al. (2007) use a pilot-by-pilot sequential approach and consider strict seniority restrictions in their model formulation. Additionally, the model of Gamache et al. (2007) includes a feasibility test that is used in the roster selection process. The feasibility test enables the program to test if there is a feasible solution to the remaining residual problem before fixing the selected roster for the considered crew member. While previous models required backtracking in the case of an encountered infeasible residual problem, with this feasibility test backtracking can be avoided. This feasiblity test is formulated as a graph coloring problem and solved using the tabu search algorithm. Gamache et al. (2007) state that the feasiblity test costs just a few minutes of computational time and that it potentially saves hours of computational time by avoiding computationally expensive backtracking. However, the model is tested on just two problem instances and even for these two instances the results are not clearly reported.

Synthesis

The Preferential Bidding System differs from equitability-based personalized rostering approaches in the way crew preferences are managed. In the Preferential Bidding System either weighted bids are taken into account in a priority ranking function to promote the preference satisfaction of senior crew members or strict seniority restrictions are adopted. Strict seniority restrictions govern that the maximization of the bidding score of a crew member can never go at the expense of the bidding score of a more senior crew member. Therefore, PBS systems under strict seniority are based on a pilot-by-pilot sequential roster assignment. Desaulniers et al. (1998) state that the Preferential Bidding System problem is even more complex than the equitability-based personalized rostering problem. Desaulniers et al. (1998) explain this notion by the fact that the Preferential Bidding System requires to solve a set-partitioning problem for each crew member in order to check whether a feasible solution can still be obtained for the remaining crew members. This significantly increases the complexity of the problem.

3. *Research Gap:* Impact of Airline Crew Rostering Elements

Academic literature on airline crew rostering has been focused on developing more efficient solution methods within a certain specific crew rostering approach. Often, research is commissioned by airlines which are interested in improving their rostering process within the framework of their rostering approach and the associated rules and regulations. Collective Labour Agreement (CLA) regulations are taken as fixed input and researchers attempt to decrease crew costs, increase crew satisfaction or increase computational efficiency within this set of prescribed constraints. However, in reality, CLA rules and regulations evolve over time through negotiations between airlines and labor unions. Besides these CLA regulations, other strategic decisions of airlines concerning the rostering process include how crew preferences are managed and what objectives are pursued. As such, a crew rostering approach should not be considered a fixed and rigid framework. Research could play an important role in providing insights on how certain *elements* of a crew rostering approach affect the efficiency and quality of the rostering process.

Definition: a *crew rostering approach* is defined as an approach to the crew rostering problem itself, i.e. a specific problem formulation.

Definition: *crew rostering elements* are defined as the aspects of a crew rostering model that constitute a crew rostering approach, such as:

(a) *Pre-assigned activities* - pre-scheduled activities such as vacations and training days;

(b) *Crew preferences* - pairings or days off preferred by crew members;

(c) *Equitability measures* - measures to create an even balance in roster characteristics;

(d) *Seniority measures* - measures to prioritize the satisfaction of senior crew members;

(e) *Roster horizon* - the number of days or weeks before operation for which a crew roster is generated;

(f) *Roster period* - the number of days or weeks of the period that is rostered.

Related research

In academic literature, just one study has been found that addresses the effect of a certain crew rostering element on the airline crew rostering process. In Gamache and Soumis (1998) three different ways of handling crew preferences have been investigated. Gamache and Soumis (1998) include crew preferences as soft constraints in the roster optimization process and compare their approach to crew preference satisfaction with two other approaches: minimum staff size and pre-assignment. The approach of Gamache and Soumis (1998) as described above is considered a compromise between the two other approaches, where in 'minimum staff size' only those crew preferences are satisfied that do not come at any cost and in 'pre-assignment' preferences are granted and pre-assigned as long there remains a feasible solution. Gamache and Soumis (1998) have shown that the compromise of including preferences in the rostering model can provide a significant increase in crew satisfaction at just a small cost in terms of roster efficiency (as shown in Figure 2.1 in Section 2.3).

A second study that can be related to the assessment of different crew rostering approaches is the paper of Maenhout and Vanhoucke (2010). The model of Maenhout and Vanhoucke (2010) was used as a simulation tool in negotiations between Brussels Airlines and the pilot union. The simulation tool was considered to be a valuable asset in the process of determining the appropriate configuration of rostering rules such as the ratio of working days versus idle days and the minimum rest between pairings. However, the results obtained in these simulation experiments have not been reported. So although no tangible insights on the effect of certain rostering rules have been presented, the research of Maenhout and Vanhoucke (2010) has confirmed that there is a demand for insights on how airlines should approach the crew rostering problem.

Research gaps

Besides addressing the impact of different approaches to handling crew preferences, Gamache and Soumis (1998) state that their approach of including preferences as soft constraints in the roster optimization model can be also be adopted for other activities, such as medical appointments, training periods and reserve blocks. These activities are usually considered pre-assigned. The integration of the rostering of these different activities into the crew rostering process is considered an interesting subject for future studies. This research gap not only concerns crew rostering elements (a) *pre-assigned activities* and (b) *crew preferences*, but should also include the (e) *roster horizon* and (f) *roster period* elements. Important questions in such a study are: when should specific type of activities be rostered and what set of activities should be rostered simultaneously?

(1) Integration of the rostering process - the assignment of pairings, vacations, training activities and reserve blocks could be integrated into one rostering process. It can be investigated whether the integration of the rostering process of these different activities results in increased roster efficiency.

Besides approaches regarding crew preferences as yet studied by Gamache and Soumis (1998), other considerations concerning crew satisfaction are (c) *equitability measures* and (d) *seniority measures*. Equitability among crew members is considered a desirable roster characteristic and many studies have included equitability measures in the development of their roster model. Similarly, many airlines prioritize the satisfaction of senior crew members over the satisfaction of junior crew members by including seniority measures in the crew rostering model. However, how these equitability and seniority measures affect the efficiency of the rostering process has not been addressed in academic literature. The future study should also include considerations on (e) *roster horzion* and (f) *roster period*, since the time horizon for which equitability or seniority is pursued is expected to have a major influence on the associated costs. In academic literature, equitability within the considered roster period is pursued. However, Kohl and Karisch (2004) state that equitability within one roster period is not the main issue, but it is equitability over the whole year what airlines and crew members are most interested in.

(2) Impact of equitability and seniority measures - equitability and seniority considerations can be taken into account as constraints or objective function coefficients in the crew rostering model. The impact of these different measures on the cost and the quality of the generated roster can be evaluated.

A third identified subject for future studies on airline crew rostering is considered to be the dynamic simulation of the rostering process. In academic literature, the airline crew rostering problem is mostly represented as a static problem that is solved at one instance in time. However, in reality the rostering process is a continuous operational process where the current roster period is influenced by previous roster periods. The model of Cappanera and Gallo (2004) has captured this effect by fixing the beginning of the roster for each crew member with the activities that resulted from the previous roster period. However, the roster process itself is still a static process in the model of Cappanera and Gallo (2004). A dynamic roster model would simulate the 'rolling horizon' or 'stepping horzion' procedure. For example, with a roster horizon of four weeks and a roster period of one week, the rolling horizon procedure would mean that every week a new 'fourth week' is added to the roster. The simulation of the rolling horzion procedure could provide insights on the following dynamic effects in the rostering process:

- *Global constraint consistency* - the capability of the rostering model to cope with constraints spanning over multiple roster periods;
- *Influx of crew requests* - crew requests come in over time and decisions on whether they should be granted depend on the rostering process;
- *Disturbances* - disturbances in crew demand and crew availability can be taken into account in the rostering process.

(3) Simulation of the dynamic rostering process - the rolling horzion procedure can be simulated. With a dynamic roster model, global constraint consistency, the influx of crew requests and the effect of disturbances can be investigated.

A final subject that is considered suitable for future studies, is the evaluation of crew rosters. Results reported in literature are often limited to the assessment of the generated rosters on the quality in terms of the objective function used in the development of the rosters. However, for the evaluation of different crew rostering approaches and methods, a way to evaluate rosters consistently and independent of the adopted objective function is required.

(4) Consistent evaluation of crew rosters - a method for the consistent evaluation of rosters can be developed. This method should take the generated rosters as input and should be independent of the approach and method adopted in the generation of the rosters.

Synthesis

Academic literature lacks a consistent evaluation of the impact of crew rostering elements on the efficiency and quality of the generated rosters. The following research gaps concerning these crew rostering elements have been identified:

1. **Integration of the rostering process** - the assignment of pairings, vacations, training activities and reserve blocks could be integrated into one rostering process. It can be investigated whether the integration of the rostering process of these different activities results in increased roster efficiency;
2. **Impact of equitability and seniority measures** - equitability and seniority considerations can be taken into account as constraints or objective function coefficients in the crew rostering model. The impact of these different measures on the cost and the quality of the generated roster can be evaluated;
3. **Simulation of the dynamic rostering process** - the rolling horzion procedure can be simulated. With a dynamic roster model, global constraint consistency, the influx of crew requests and the effect of disturbances can be investigated;
4. **Consistent evaluation of crew rosters** - a method for the consistent evaluation of rosters can be developed. This method should take the generated rosters as input and should be independent of the approach and method adopted in the generation of the rosters.

4. *Benchmark:* Approaches to Crew Rostering in Other Industries

The airline crew rostering problem belongs to the general class of problems called personnel scheduling (Jarrah and Diamond, 1997). In order to have a deeper understanding of the airline crew rostering problem, a benchmark on personnel scheduling approaches in other industries is performed. First, an overview of the generic personnel scheduling problem and its characteristics is presented. Afterwards, the literature on personnel scheduling related to the most relevant industries is discussed separately. The public transit industry is chosen for its comparability with the airline crew rostering problem. The healthcare industry is considered because the nurse rostering problem is the most studied personnel scheduling problem in academic literature (Bergh et al., 2013). Finally, the personnel scheduling problem in the call centre industry is discussed.

4.1 Generic Personnel Scheduling

In order to give some context to the airline crew scheduling problem, some generic concepts in personnel scheduling will be discussed. First, various classifications of types of personnel scheduling problems will be discussed. Afterwards, the components of the personnel scheduling problem will be presented. Lastly, a generic mathematical model for the personnel scheduling problem will be presented. Each section will also briefly discuss the application of these general personnel scheduling concepts to the airline crew scheduling problem.

Types of Personnel Scheduling Problems

An early classification of personnel scheduling problems was proposed by Baker (1976). Three types of scheduling problems can be distinguished:

- *Shift* scheduling;
- *Days-off* scheduling;
- *Tour* scheduling.

The shift scheduling problem arises in organizations where operations run longer than the regular eight-hour workday. The shift scheduling problem is a daily problem and addresses which eight-hour shift should be assigned to each employee. In organizations that run for the regular five working days of the week, i.e. Monday to Friday, the shift scheduling problem is the only problem that needs to be solved (Morris and Showalter, 1983). For organizations running a different workweek, e.g. seven days a week, the days-off scheduling problem needs to be solved in order to determine which days of the week each employee will be working. If both the shift scheduling and the days-off scheduling problems arise, the combined integrated problem is called the tour scheduling problem. The tour scheduling problem often addresses operations that run seven days a week and having more than one shift a day. A typical simplified problem is presented by Morris and Showalter (1983), where each employee shall be assigned a tour of five consecutive work days with a daily shift beginning at the same time each day. They considered shift starting times to be any hour of the day. As such, 168 possible tours could be constructed, i.e. 24 times 7 tour starting times. Each employee then has to be assigned one of these 168 tours such that the demand is covered.

A second classification found in literature, is the following distinction:

- *Cyclic* or *cyclical* scheduling;
- *Non-cyclic* or *non-cyclical* scheduling.

There are two aspects of the scheduling problem that can be cyclic or cyclical (considered synonyms). Warner (1976) defines the *cyclical* approach as the effort to construct an initial high-quality schedule, consisting of a roster for each employee for the complete roster period, which is then repeated period after period. A different interpretation of *cyclicity* is adopted in a comprehensive review of personnel scheduling problems of Ernst et al. (2004b). Ernst et al. (2004b) considers a *cyclic* schedule to be a schedule in which every employee essentially has the same roster, but with different starting times for the first shift or duty. Because of of its inflexibility to changing environments such as changing demands, the cyclical scheduling problem as considered by Warner (1976) has become obsolete. However, cyclic personnel scheduling as considered by Ernst et al. (2004b) is still being adopted and studied in academic literature (Musliu et al., 2002; Rocha et al., 2013; Xie and Suhl, 2014). In this literature study, *cyclic* scheduling or rostering will therefore refer to cyclic schedules with employees having the same rosters only lagged in time (Xie and Suhl, 2014).

De Causmaecker et al. (2004) propose a different classification of personnel scheduling problems which focuses more on the demand profile for which the personnel scheduling problem is solved. Four different types of personnel scheduling problems are identified:

- *Permanence centred* planning;
- *Mobility centred* planning;
- *Fluctuation centred* planning;
- *Project centred* planning.

The *permanence centred* planning problem appears in police services and in hospitals. The number of personnel that is needed is known beforehand, except for emergency cases. Nurse rostering, which will be discussed into more detail in Section 4.3, is a typical example of permanence centred planning. *Mobility centred* planning appear in companies that involve transportation for the duties to be carried out. These problems occur in for example home health care organizations. Pure transportation problems are excluded in the analysis of De Causmaecker et al. (2004). However, in a later paper, Brucker et al. (2011) explicitly classify the crew scheduling problem faced by transportation companies like airlines and public transit companies as a mobility centred planning problem. *Fluctuation centred* planning problems appear in industries that are faced with fluctuating demand. The call centre industry is the most typical example of an industry with fluctuation centred planning problems. Finally, there are companies that adopt *project centred* planning, by dividing the work into projects that need to be assigned to different groups of employees.

Relating the above analysis to the airline crew scheduling problem, it can be stated that the airline crew scheduling problem is a *mobility centred tour* scheduling problem. Furthermore, airline crew scheduling should be approached as a *non-cyclic* scheduling problem. *Cyclic* scheduling is considered inappropriate for airline crew scheduling problems, since the demand fluctuates over time and the ”shifts”, i.e. the pairings, have different lengths and starting times (Ernst et al., 2004b). Moreover, cyclic schedules cannot take individual personalized restrictions or preferences into account (Bourdais et al., 2003).

Stages in Personnel Scheduling

Besides context of different types of personnel scheduling problems, academic literature also provides frameworks for how to approach the personnel scheduling problem. Tien and Kamiyama (1982) proposed a first decomposition of the generic personnel scheduling problem into five separate subproblems. Ernst et al. (2004b) provided a comparable but more comprehensive framework. This framework, consisting of six separate modules:

1. *Demand modelling* is aimed to determine how many employees are needed at different times across the rostering horizon;
2. *Days-off scheduling* is done to determine, based on the modeled demand and the available employees, how many employees should have a day-off at any day in the roster period;
3. *Shift scheduling* concerns the timing of the shifts and the number of required workers per shift;
4. *Line of work construction* is the process of constructing roster lines consisting of shifts and days-off for the complete roster horizon;
5. *Task assignment* is sometimes necessary to determine which tasks need to be carried out during certain shifts since certain tasks may require particular staff qualifications which have to be taken into account in the final module;
6. *Staff assignment* is the last step of the process and involves the assignment of a roster to each employee.

Ernst et al. (2004b) distinguish three types of demand: *task based demand*, *flexible demand* and *shift based demand*. Crew scheduling problems in the transportation industry are examples of problems that are subject to *task based demand*. The *demand modelling*, *shift scheduling* and *task assignment* modules are then integrated as a *duty generation* module, which involves the aggregation of individual tasks into larger pieces of work (Ernst et al., 2004b). The crew pairing problem as appearing in the airline industry is a typical example of duty generation (Brucker et al., 2011; Ernst et al., 2004b). The other subproblem of the airline scheduling problem, the crew rostering problem, then can be considered to be integrating the *days-off scheduling*, *lines of work construction* and *staff assignment* modules. More specifically, with the bidlines approach the lines of work are constructed first and then assigned to the staff whereas with personalized rostering approaches these two final modules are completely integrated.

Generic Mathematical Model for Personnel Scheduling

Brucker et al. (2011) provide a generic mathematical model for personnel scheduling. Brucker et al. (2011) model the problem as a binary linear program:

$$min \sum_{e \in E} \sum_{\pi \in P_e} c_{e\pi} x_{e\pi} \tag{4.1}$$

$$\sum_{\pi \in P_e} x_{e\pi} \leq 1, \qquad e \in E \tag{4.2}$$

$$\sum_{e \in E} \sum_{\pi \in P_e} \pi(j,t) \cdot x_{e\pi} \geq D_j(t), \qquad \forall (j,t) \tag{4.3}$$

$$x_{e\pi} \in \{0,1\} \qquad \forall e \in E \quad \text{and} \quad \pi \in P_e \tag{4.4}$$

- The planning horizon $[0, T]$ is divided into periods $[t, t+1]$ for $t = 0, 1, ..., T-1$ for which tasks $j = 1, ..., m$ must be performed.
- The demand profile for task j,$D_j(t)$, is the number of employees needed to perform task j in time period $[t, t+1](t = 0, 1, ..., T-1)$.
- Associated with each employee $e \in E$ is a subset Q_e of tasks for which e is qualified, i.e. e can be assigned to tasks in Q_e only.
- A working pattern is defined as $\pi = (\pi(j, t))$, where $\pi(j, t) = 1$ if and only if $[t, t+1]$ is a working period in which task j has to be performed.
- $x_{e\pi}$ is a binary decision variable which is one if and only if the working pattern π is assigned to employee e.
- $c_{e\pi}$ is the costs associated with the assignment of employee e to working pattern π.

The objective function aims to minimize the total cost of all working patterns. Constraint 4.2 makes sure that at most one working pattern is assigned to each employee. If each employee must be assigned a working pattern, the inequality $\leq$ should be replaced by an equality sign. Constraint 4.3 forces the number of employees assigned to task j per period $[t, t+1]$ to be at least equal to the demand of j in period $[t, t+1]$. Finally, 4.4 is the expression for the requirement of having binary decision variables in the model solution.

The mathematical model of Brucker et al. (2011) is a basic model that can be applied to various different personnel rostering problems and appended with additional objective function components and constraints. Bergh et al. (2013) provides an extensive overview of the different types of hard and soft constraints that can be found in the academic literature on personnel scheduling. As can be seen in the generic model presented above, the model implies that the input contains a set of working patterns $\pi(j, t)$, i.e. a sequence of tasks. The roster optimization model thus concerns the roster selection process. However, the generation of the working patterns that takes place before the actual roster selection is also an important part of the personnel rostering process. When applying this generic model to the airline industry, a task can be considered a flight and a working pattern then relates to the sequence of pairings that constitutes a crew roster. The demand profile $D_j(t)$ then is the demand for pilots per flight segment j which follows directly from the flight schedule for the considered roster period.

4.2 Public Transit Industry

The crew scheduling problem encountered in the public transit industry, i.e the railway, urban mass transit and bus industries, is very comparable to the airline scheduling problem. Both are mobility centred planning problems, in which the demand follows directly from a given timetable and tasks involve both temporal and spatial features (De Causmaecker et al., 2004; Ernst et al., 2004b). Similar to what is observed in the airline industry, North American and European approaches to the public transit crew rostering problem are completely different (Carraresi and Gallo, 1984; Desaulniers and Hickman, 2007). In North America, the rostering process is left to the drivers themselves and selection is made on the basis of seniority. In Europe on the other hand, union conventions prescribe that an even balance of workload should be ensured over a certain time horizon. Since in North America crew members build their own rosters in order of seniority, there is no actual crew rostering optimization. Therefore, literature on public transit crew rostering is focused on the European problem of optimizing for minimum cost and for equitability.

Heuristic solution techniques

Two surveys on public transit crew rostering, of Odoni et al. (1994) and Desaulniers and Hickman (2007), state that the common practice for the transit crew rostering problem is using a constructive heuristic to develop an initial feasible solution and then improving this initial solution by using an improvement heuristic. The initial solution is constructed by formulating an assignment problem for each day of the horizon, in order to assign the duties of the corresponding day to the partial rosters that were built by the previous assignment problems (Desaulniers and Hickman, 2007). The improvement heuristic used afterwards is mostly based on exchange procedures (Odoni et al., 1994). An example of such a solution methodology was proposed by Carraresi and Gallo (1984). They formulate the bus drivers rostering problem as a Multi-level Bottleneck Assignment (MBA) problem, which is aimed at maximizing the quality of the roster that is the worst from the drivers' perspective. In order to do so, each shift is given a "weight" which represents the workload of the shift for the drivers and encompasses factors such as the length and timing of the shift. The objective function then is to minimize the maximum total weight of the shifts assigned to a single driver. Carraresi and Gallo (1984) solve the bus drivers rostering problem using a so-called 'asymptotically optimal algorithm', which first constructs a feasible solution where after this initial feasible solution is improved. The algorithm is developed to seek successive improvement by identifying 'bottleneck arcs', i.e. shift assignments with unsatisfactory high weights. For such a bottleneck arc the algorithm tries to find an alternative assignment by searching for a 'decreasing alternating chain' of arcs that does not contain the bottleneck arc. By doing so, the overall weight of the roster is decreased. Bianco et al. (1992) continue the work of Carraresi and Gallo (1984) by proposing an improved version of this Multi-level Bottleneck heuristic algorithm. This new algorithm of Bianco et al. (1992) was found to outperform the algorithm of Carraresi and Gallo (1984) in terms of solution quality. However, since the new algorithm required significantly more computational time, Bianco et al. (1992) conclude that their algorithm should just be used for problems with a roster period of up to 10 days.

Caprara et al. (1998) also adopt a heuristic approach to the public transit crew rostering problem. However, the heuristic as described in Caprara et al. (1998) constructs the crew rosters directly. As such, the solution method of Caprara et al. (1998) is a constructive heuristic. A refinement procedure is included afterwards to further improve the outcome, but this refinement procedure is based on the same constructive heuristic algorithm which is applied on a number of duties. The constructive heuristic of Caprara et al. (1998) constructs rosters sequentially for each crew member, as opposed to the day-by-day approach adopted by Carraresi and Gallo (1984) and Bianco et al. (1992).

Desaulniers and Hickman (2007) state that these heuristic approaches to the public transit crew rostering problem are still in use because of their computational speed and flexibility in dealing with constraints following from union regulations. This is in contrast to solution methods adopted in airline crew rostering, which are mostly based on mathematical programming solution techniques. Desaulniers and Hickman (2007) explain this difference by stating that public transit crew rostering problems are even larger than the airline crew rostering problem. The transit problems are not separable per vehicle type, since drivers are often allowed to drive all vehicle types. Also, in airline crew rostering the roster blocks correspond to pairings that span several days while in public transit the crew rostering problem involves rostering individual duties. Desaulniers and Hickman (2007) therefore argues that the combinatorial complexity of the public transit crew rostering problem is too large for mathematical programming methods and that heuristic approaches are preferred.

Alternative solution techniques

However, probably due to advances in computer processing capabilities, in recent years Lin Xie and Leena Suhl have developed crew rostering models based on different solution methods than the traditional constructive and improvement heuristics. The paper of Xie and Suhl (2014) describes a mathematical programming model based on a multi-commodity flow network. The model of Xie and Suhl (2014) solves both the cyclic and the non-cyclic crew rostering problem. Moreover, the paper of Xie and Suhl (2014) was the first to include crew preferences in the non-cyclic crew rostering model. The personalized model of Xie and Suhl (2014) for the non-cyclic problem aims for minimum cost, for equitability among drivers and for crew preference satisfaction. The multi-commodity flow problem is solved using the IBM CPLEX optimizer. Although most problems are solved within acceptable run times, some larger problem instances could not be solved in 24 hours. Because of these unsatisfactory run times, in Xie et al. (2017) a solution method based on meta-heuristics is proposed. Three meta-heuristics, ant colony optimization, simulated annealing, and tabu search methods, are modeled and their outcomes are compared. The simulated annealing meta-heuristic was found to be superior to the other meta-heuristics. Moreover, for large problem instances the simulated annealing algorithm also outperforms the IBM CPLEX solver.

Synthesis

In public transit crew rostering three approaches can be distinguished. First there is a distinction between cyclic and non-cyclic crew rostering (see Subsection 4.1 for explanation). Two non-cyclic approaches are then found in literature, being the North American approach and the European approach. In North America crew members select their own rosters in order of seniority so there is no actual roster optimization. In Europe, the crew rostering approach in the public transit industry is very comparable to the equitability-based personalized rostering approach in the airline industry. The objective is not only to minimize costs but also to ensure an even balance in roster characteristics among crew members. In terms of solution techniques, most models are based on a combination of constructive and improvement heuristics.

4.3 Health Care Industry

In academic literature on crew rostering in the health care industry, the main crew rostering problem that is addressed is the nurse rostering problem (Ernst et al., 2004b). It is a permanence centred planning problem and having appropriate levels of staff in the different medical wards is essential (De Causmaecker et al., 2004; Ernst et al., 2004b). Other crew rostering problems encountered in the health care industry are the rostering of home health care personnel and the rostering of personnel for the handling of hospital call arrivals (Ernst et al., 2004b). However, this section of the literature study specifically addresses the nurse rostering problem, since academic research on this subject is well developed and the related best practices could provide valuable insights for future studies into airline crew rostering.

Nurse rostering approaches

In an early influential work, Warner (1976) distinguishes two generic scheduling approaches: the *traditional approach* and the *cyclical approach*. In the cyclical approach one schedule is constructed for an initial rostering period and this schedule is then repeated period after period. The traditional approach on the other hand can be called non-cyclical since every rostering period is considered independently and resources are spent without pursuing repeating patterns. Millar and Kiragu (1998) propose a model based on network programming that can accommodate both approaches, called *cyclic* and *non-cyclic* scheduling by Millar and Kiragu (1998). Cyclic schedules are considered to be more stable and perceived as more fair by the nurses according to Millar and Kiragu (1998). Also, the cyclic problem is simpler than the non-cyclic problem, resulting in less scheduling effort and required computational power. However, the disadvantages of cyclic schedules are their rigidity, resulting in difficulties when adapting cyclic schedules to changing circumstances.

Silvestro and Silvestro (2000) make a similar distinction in nurse rostering approaches, but use a different terminology: *fixed rostering* and *flexible rostering*. Fixed (cyclic) rostering is considered inappropriate for the nurse rostering problem and the paper thus focuses on flexible (non-cyclic) rostering. Silvestro and Silvestro state that flexible rostering can be further subdivided into three approaches based on the degree of self-control a nurse has about his or her roster. The following flexible rostering approaches are distinguished: *departmental rostering*, *team rostering* and *self-rostering* (Silvestro and Silvestro, 2000). Departmental rostering is the approach where the rostering is conducted by a manager and as such can be considered centralized. Team rostering on the other hand divides the complete pool of staff into teams, where each team has the responsibility for the rostering of the staff members in their team. Finally, self rostering is done by the staff members themselves. The three nurse rostering approaches are extensively discussed and compared in their paper. However, their evaluation of the rostering approaches are based on a qualitative and empirical study, consisting of telephone interviews and practical case studies. Although their study does present a good overview of benefits and limitations of the three approaches, the outcomes are limited to qualitative statements rather than quantitative results and comparisons. Still, it can be concluded from the comparative study of Silvestro and Silvestro (2000), that for large ward sizes with many nurses (>70 staff) self-rostering and team rostering are ineffective and departmental rostering is the most suitable nurse rostering approach.

Cheang et al. (2003) also recognize the two basic types of scheduling, calling them *cyclic* and *non-cyclic* scheduling. Furthermore, they consider three different *views* of the nurse rostering problem; a *nurse-day view*, a *nurse-task view* and a *nurse-shift pattern view* (Cheang et al., 2003; Santos et al., 2016). These different views result in different kinds of decision variables as illustrated by equations 4.5, 4.6 and 4.7 respectively.

$$v_{ijk} = \begin{cases} 1 & \text{nurse } i \text{ works shift } k \text{ on day } j \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

$$v_{is} = \begin{cases} 1 & \text{nurse } i \text{ receives task } s \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

$$v_{ip} = \begin{cases} 1 & \text{nurse } i \text{ works shift pattern } p \\ 0 & \text{otherwise} \end{cases} \tag{4.7}$$

Valouxis et al. (2012) adopted an approach that can be considered to be a hybrid form of two of the approaches discussed by Cheang et al. (2003). Valouxis et al. (2012) decompose the problem into two parts. First, all nurses are assigned to specific days of the schedule, based on the demand per day and the availability of the nurses. Afterwards, on each specific day the actual shifts are divided among the nurses that were appointed to that day in the first phase.

Another set of nurse rostering approaches is found in the paper of Bard and Purnomo (2005). Bard and Purnomo (2005) distinguish three nurse rostering approaches: *traditional* or *cyclical scheduling*, *self-scheduling* and *preference scheduling*. Preference scheduling is considered a combination of the traditional approach and the self-scheduling approach since the rostering process itself is centralized but individual nurse members can express their preferences for certain aspects of the roster. As such, it is a personalized departmental rostering approach and similar to the personalized rostering approaches encountered in the airline industry. Interestingly, whereas in an early work of Warner (1976) non-cyclical rostering is called the traditional approach, about thirty years later it is the cyclical approach that is called *traditional*. Bard and Purnomo (2005) reject the cyclical approach and self-scheduling approach for their rigidity and impracticality respectively. They therefore consider preference scheduling to be the most promising approach. In two related papers, Asgeirsson explores the possibility of combining two of the rostering approaches of Bard and Purnomo (2005): self-scheduling and preference scheduling (Asgeirsson, 2014; Asgeirsson and Sigurdardottir, 2016). In this approach, each nurse is responsible for creating a good preliminary schedule, which is then considered input for the preference scheduling as done by a personnel manager.

Solution techniques

In terms of solution methodologies, academic literature provides an extensive range of possibilities. In a comprehensive literature review, Burke et al. (2004) discusses nurse rostering studies with the following solution methods: mathematical programming, goal programming, artificial intelligence methods, heuristics and meta-heuristics. Burke et al. (2004) evaluate that mathematical programming methods cannot cope with the enormous problem sizes encountered in real problems. However, heuristics cannot cope with the dynamic and uncertain nature of realistic problems according to Burke et al. (2004). Therefore, Burke et al. (2004) argue that hybrid methods between exact methods and heuristics are the only realistic way to solve such big combinatorial problems. The state-of-the-art in nurse rostering therefore is focused on complex hybrid solution methods, combining mathematical programming and heuristics with other search methods such as meta-heuristics and constraint programming (Burke et al., 2004).

Recent contributions

Current research is focused on making nurse rostering models more realistic by incorporating practical operational characteristics of the nurse rostering problem. Previous research has considered the nurse rostering problem in a *static horizon*, considering only the roster period for which the roster was generated. Salassa and Vanden Berghe (2012) however proposed a different approach, addressing the nurse rostering problem in a *stepping horizon* context, where rosters from preceding roster periods are taken into account during the rostering process. The Second International Nurse Rostering Competition (INRC-II) was aimed to challenge researchers to develop nurse rostering models with a stepping horizon. Smet et al. (2016) evaluate the stepping horizon model and compare it to the traditional approach of using a static horizon. Smet et al. (2016) explicitly advocate the use of a stepping horizon and provide guidelines on how to ensure global consistency in constraints over the different roster periods. Furthermore, Smet et al. (2016) propose that future research efforts should aim at developing models that can also take future roster periods into account, by modelling uncertainty through stochastic programming or by taking a-priori known future events into account in the current rostering period.

A recent contribution made by Mihaylov et al. (2016) concerned the learning of preference soft constraint weights from historical data. With preference scheduling, like with personalized rostering approaches in the airline industry, nurses express preferences on certain tasks or shifts. These preferences can be taken into account as hard constraints, but most models deal with preferences as soft constraints, by either rewarding the satisfaction of preferences or by penalizing for not doing so. However, the weights of these rewards or penalties in the objective function often are based on trial-and-error and as such are considered quite naive. Mihaylov et al. (2016) propose an alternative methodology to construct these soft constraint weights, by automatically extracting these weights from previous rosters.

Synthesis

Various nurse rostering approaches have been found in academic literature. For larger complex problems that are more comparable with problems encountered in the airline industry, *non-cyclic departmental rostering* is considered the only realistic rostering approach. *Preference scheduling* is considered a specific type of departmental rostering approach, where nurses can express their preferences but where the actual rostering process is still a centralized process. Researchers adopt various solution methodologies in their models, but hybrid combinations between exact methods, heuristics and meta-heuristics are considered to be most promising. Finally, recent advances have made nurse rostering models more realistic by considering the problem in the context of a *stepping horizon* and by leveraging historical data in the development of the model.

4.4 Call Centre Industry

Gans et al. (2003) provide an overview of call centre operations and one of the discussed aspects is the call centre personnel scheduling problem. Gans et al. (2003) state that the call centre personnel scheduling problem is divided into two separate problems: a scheduling problem and an assignment problem. Basically, the scheduling problem aims at constructing a set of anonymous rosters that cover all shifts. Then, the assignment problems deals with assigning these rosters to individual employees. As such, this approach can be compared to the bidline scheduling approach in the airline industry.

Aksin et al. (2007) state that the objective of the shift scheduling and rostering problem in the call centre industry is to minimize personnel costs while achieving service levels or other labour requirements. The service levels constraints follow from the staffing problem, which deals with the problem of translating the stochastic demand to the number of personnel required per shift. Aksin et al. (2007) discusses the issues of solving the call centre rostering problem using mathematical programming, explaining why most researchers propose call centre models are based on heuristics (Avramidis et al., 2009; Cezik and L'Ecuyer, 2008; Fukunaga et al., 2002). In terms of rostering approaches, Aksin et al. (2007) state that the call centre personnel scheduling problem is usually divided between a scheduling problem and an assignment problem. However, a major issue with this sequential approach is discussed, being its inflexibility to disturbances and updates in call forecasts. Therefore, it is stated that many large call centres adopt a different approach, called "*shift bidding*". This approach is very comparable to the Preferential Bidding System (PBS) encountered in the airline industry. In shift bidding, call centre employees can bid on particular shifts sequentially, with the order of bidding based not only on seniority but also on previous quality of service delivered by the employee (Aksin et al., 2007).

An example of a model that adopts such a shift bidding approach is presented in Hojati (2010). In their model, employees can express their preferences for the shifts they would want to work. The deviation of the starting time of the assigned shift from the starting time of the preferred shift is then minimized. However, the personnel scheduling problem in call centres is considered to be less complicated than the crew scheduling problem encountered in airline industry, since only one shift per day needs to be assigned to the available staff. Also, there are far less regulations and constraints that need to be taken into account in the rostering process. Similar to with the Preferential Bidding System for airline scheduling, there is an integer linear program that needs to be solved for every employee, in order of seniority. Hojati (2010) state that these separate ILP problems are too complicated to be solved with an exact method and therefore uses a constructive heuristic based on sequential assignment of shifts to employees.

5. *Methodology:* Crew Rostering Solution Methods

This chapter will discuss various modelling techniques that can be used in future studies on airline crew rostering. The first step in constructing a model is translating the actual problem to a mathematical model. The different variations encountered in airline crew rostering literature will be discussed in Subsection 5.1. To capture the time-space constraints associated with the activities in a roster model, researchers often represent the model on a network. Some important aspects of this network representation will be discussed in Subsection 5.2. There are various techniques that can be used to solve an airline crew rostering problem. Some important solution techniques will be explained in Subsection 5.3.

5.1 Model Formulation

The airline crew rostering problem is a *combinatorial optimization problem*, and belongs to the class of NP-complete problems (Deveci and Demirel, 2018) and as such also an NP-hard problem (Lucic and Teodorovic, 2007). In essence, there are two main groups of solution methodologies to the crew rostering problem: heuristics and mathematical programming methods (Bergh et al., 2013). However, many researchers adopt hybrid methods, combining mathematical programming techniques with (meta-)heuristics to solve the problem. Mathematical programming methods encountered in literature are based on integer programming and often adopt a set-partitioning problem formulation (Deveci and Demirel, 2018; Kasirzadeh et al., 2017). Furthermore, some research papers address the crew rostering problem in an operational and dynamic context, and as such include stochastic elements in the model formulation. This subsection will provide some background on these various mathematical programming methods, in order to give insight into how the crew rostering problem could be modelled.

Integer Linear Programming (ILP)

An Integer Linear Programming (ILP) model is the linear programming model with the additional restriction that the decision variables must have integer values (Hillier and Lieberman, 2015). In a crew rostering model the decision variable values often are restricted even more and the model is in the form of a Binary Integer Programming (BIP) model, meaning that decision variables must be either a 0 or a 1. In the case of the airline crew rostering problem, the decision variables then become (Deveci and Demirel, 2018):

$$x_{ir} = \begin{cases} 1 & \text{if crew member } i \text{ receives roster } r \\ 0 & \text{otherwise} \end{cases} \tag{5.1}$$

Set-partitioning problem

The *set-partitioning problem* is a special type of integer programming problem that is formulated as follows (quoting Balas and Padberg (1976)):

”*Given:*

- *a finite set M;*
- *a constraint set defining a family F of 'acceptable' subsets of M; and*
- *a cost (real number) associated with each member of F;*

find a minimum-cost collection of members of F which is a partition of M”

In an airline crew rostering problem, the goal is to generate a roster for each crew member such that all pairings that need to be flown are covered (Kasirzadeh et al., 2017). As such, the problem can be considered a *set-partitioning problem*, since the set of pairings need to be covered by a set of rosters. For the airline crew rostering the set-partitioning problem can be expressed mathematically as follows:

$$\sum_{j=r}^{R} a_{pr} \cdot x_{ir} = 1, \qquad \forall j \in P \tag{5.2}$$

$$a_{pr} = \begin{cases} 1 & \text{if pairing } p \text{ is covered by roster } r \\ 0 & \text{otherwise} \end{cases} \tag{5.3}$$

Stochastic programming

Traditional airline crew scheduling models consider the deterministic problem without considering potential disruptions (Deveci and Demirel, 2018). Some recent papers do address these stochastic factors and aim to generate robust crew schedules, capable of withstanding these disruptions. Rosenberger et al. (2002) developed a stochastic model for airline operations based on a discrete event semi-Markov process and simulates disturbances using Monte Carlo sampling. The considered operational recovery instruments are delays, cancellations, deadheads, plane ferries, reserve crews, reconstructed pairings and rerouting passenger itineraries. Ionescu and Kliewer (2011) present a stochastic crew scheduling model that incorporates crew swap opportunities between pairings during the crew scheduling process. Ingels and Maenhout (2018) evaluate the impact of proactively scheduling of overtime as a buffer in the crew scheduling problem.

An important distinction is made between *stochastic programming* and *robust optimization*. In stochastic programming, the goal is to optimize the *expected value* of the objective function and as such aims at providing solutions that perform well on average (Hillier and Lieberman, 2015). Robust optimization on the other hand avoids using probability distributions and focuses on providing a conservative solution that can handle the worst possible scenario (Hillier and Lieberman, 2015). Hillier and Lieberman (2015) state that stochastic programming is more appropriate for problems with soft constraints while robust optimization is better suited for problems with hard constraints.

In stochastic programming, models can be developed that addresses problems in a dynamic context, where certain decisions can be delayed until later in the process. This is called *stochastic programming with recourse*, and with such a model decisions can be delayed such that some or all of the uncertainties in the problem are eliminated (Hillier and Lieberman, 2015). Also, corrective actions can be made later in the process in the case that initial decisions show to have undesirable consequences. A *two-stage problem* for example is a stochastic programming with recourse problem where decisions are made in two stages. *Multistage problems* consider multiple stages where decisions are made as uncertainty is eliminated from the model over time (Hillier and Lieberman, 2015).

Dynamic modelling

While current state-of-the-art airline crew rostering models only consider the problem in a static context, recently Smet et al. (2014) proposed a dynamic crew rostering model for the nurse rostering problem. The dynamic crew rostering model has a *stepping horizon* instead of a fixed roster horizon. With a roster horizon of four weeks and a roster period of one week, the stepping horizon procedure entails that every week a new 'fourth week' is added to the roster. As such, rostered activities from preceding roster periods are taken into account as restrictions for the current roster period. An observation made by Smet et al. (2014) concerns the fact that in the operational environment a-priori known future events could also be incorporated in the crew rostering process. For example, there might be pre-assigned crew activities such as holidays or training activities already rostered to take place in future roster periods which could be taken into account in the current roster period.

5.2 Network Representation

In the modelling of the airline crew rostering problem, researchers often use network flow models (Ernst et al., 2004a; Kasirzadeh et al., 2017). The airline crew rostering problem is very suitable for such a network flow model since the activities that need to be rostered are described in time and space. Also, crew members can be seen as *resources* or *commodities* that need to be distributed over the network of pairings (Cappanera and Gallo, 2004). This section discusses the main concepts of network flow models.

Minimum cost multi-commodity flow problem

The airline crew rostering problem is an example of the general assignment problem, and as such is a special type of the *minimum cost multi-commodity flow problem* (Hillier and Lieberman, 2015). The minimum cost flow problem is the most general network model and encompasses many other specific problems such as the *shortest path problem*, the *minimum spanning tree problem* and the *assignment problem*. The minimum cost flow problem considers flow through a network with limited arc capacities (Hillier and Lieberman, 2015). The objective in the minimum cost flow problem is to minimize the total cost of sending the available commodities through the network to satisfy the given demand (Hillier and Lieberman, 2015). The airline crew rostering problem is a *multi-commodity flow problem* and each crew member corresponds to a commodity (Cappanera and Gallo, 2004).

Time-space network

A roster of a crew member can be represented on a path along a set of pairings defined on a directed acyclic *time-space network*. For every crew member, a *resource constrained shortest-path* needs to be found, alternating between *activity arcs* such as pairing arcs and *compatibility arcs* that link together pairs of activities that can be assigned consecutively (Cappanera and Gallo, 2004; Kasirzadeh et al., 2017). The resource constrained shortest-path problem then can be solved using various solution algorithms such as dynamic programming, labelling algorithms and constraint programming (Irnich and Desaulniers, 2005). In Figure 5.1, an example of a time-space network representation can be found.

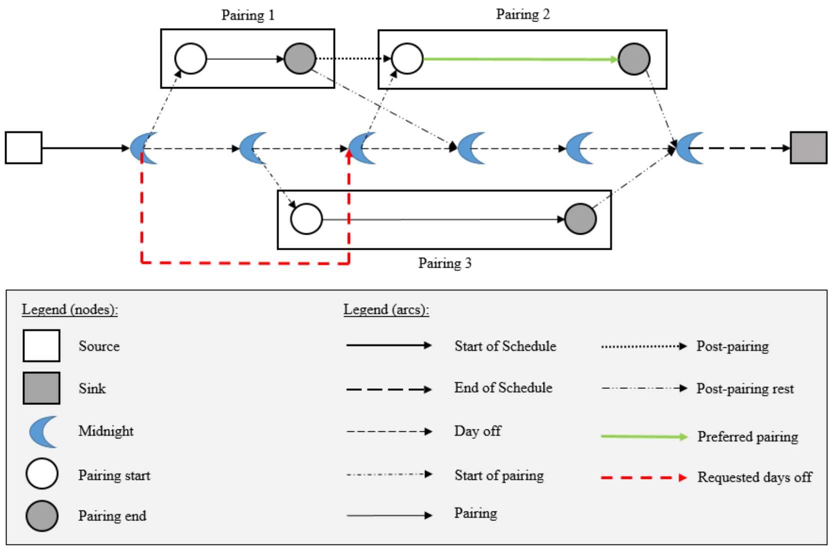

Figure 5.1: Example of a directed acyclic time-space network representation of a personalized crew rostering subproblem (based on Saddoune et al. (2012) and Kasirzadeh et al. (2017)).

5.3 Solution Techniques

In academic literature, many different solution methodologies are adopted in the modelling of the airline crew rostering problem. A distinction can be made between mathematical programming techniques and heuristic techniques, where mathematical programming focuses on finding the optimal solution and heuristics aim for finding a feasible sub-optimal solution in reasonable computational time (Burke and Kendall, 2005). This subsection addresses some of the main solution techniques that are used to solve the airline crew rostering problem.

Mathematical Programming

Mathematical programming techniques are aimed to find the optimal solution to a given optimization problem. This section discusses some important mathematical programming techniques that are often used in airline crew rostering.

Linear Programming relaxation

Integer Linear Programming (ILP) and Binary Integer Programming (BIP) models may seem 'easier' to solve than general Linear Programming (LP) models, since only the integer or binary solutions have to be considered. However, the opposite is true (Hillier and Lieberman, 2015). The linear characteristics of LP models can be exploited very efficiently by LP algorithms such as the simplex method. Therefore, large ILPs and BIPs are often solved by relating the ILP problem to the corresponding LP problem. This corresponding LP problem is called the *Linear Programming (LP) relaxation* of the considered ILP problem. The LP relaxation is the same problem as the ILP problem, except that the restriction for integer solutions is removed (Hillier and Lieberman, 2015). The solution to the LP relaxation of the problem then provides an *upper bound* to the solution of the actual ILP problem. However, for some special types of ILP problems it is guaranteed that the solution of the LP relaxation is integer and thus the optimal integer solution.

Lagrangian relaxation

Another type of relaxation of a mathematical programming problem is the *Lagrangian relaxation*. The Lagrangian relaxation of an LP (or ILP or BIP) problem removes the functional constraints from the model and adds them in the objective function with a Lagrange multiplier that penalizes if a constraint cannot be satisfied (Hillier and Lieberman, 2015). The Lagrangian relaxation of an LP problem can be solved extremely efficiently, but similar to the LP relaxation only provides a tight upper bound to the optimal solution value.

Branch-and-bound

Since the airline crew rostering problem is a bounded integer problem, there are only a finite number of feasible solutions. Therefore, mathematical programming methods used to find the optimal solution are often based on an enumeration procedure (Hillier and Lieberman, 2015). However, for large problems enumerating all possible solutions can be very computationally expensive. Therefore, mathematical programming techniques are used to guide which solutions are considered in the enumeration procedure. An efficient way to structure such an enumeration procedure is by dividing the problem into smaller sub-problems and then discarding some of the sub-problems that have sub-optimal solution sets. This procedure is called the *branch-and-bound* technique.

The branch-and-bound scheme can be represented by a solution tree, where the overall problem is subdivided into smaller and smaller sub-problems. Sub-problems or *branches* are explored and the *bounds* on the optimal solution within such a sub-problem is determined. Lagrangian relaxation is often used in this bounding procedure, since it can quickly provide an estimation of the bounds of the optimal solution within the sub-problem (Hillier and Lieberman, 2015). These bounds are then compared to the optimal solution found by the algorithm so far in another sub-problem. If the current optimal solution is better than the bounds of the sub-problem under consideration, it can be concluded that no better solution can be found in the entire sub-problem. As such, the sub-problem can be discarded and the complete enumeration of all possible solutions within the sub-problem is avoided.

Column generation

Another method to structure and reduce the mathematical enumeration procedure is by using *column generation*. In column generation, the model starts with an initial subset of variables that permits an initial feasible solution. Afterwards, this initial feasible solution is improved by identifying additional variables that have reduced costs such that an improved LP solution can be achieved (Hooker and van Hoeve, 2018). This process is iterated until there are no variables left that when added to the model would improve the solution. Then, it can be concluded that the current solution is optimal. This procedure is called *column generation* since variables correspond to columns in the matrix representation of an LP model (Hooker and van Hoeve, 2018). At any point in the iterative process, the current LP model with the current set of variables is called the *Restricted Master Problem (RMP)*. The procedure of identifying the variables that need to be added to the RMP is called the *pricing problem*.

In the airline crew rostering problem, the decision variables in the set-partitioning problem formulation correspond to a roster for a crew member (Deveci and Demirel, 2018). As such, in the column generation scheme finding a new column to be added to the RMP corresponds to finding a resource-constrained shortest path with reduced cost (Kasirzadeh et al., 2017). Often, the column generation technique is embedded in a branch-and-bound tree. As such, every sub-problem in the brand-and-bound tree is solved using a column generation procedure (Ernst et al., 2004a). This technique that combines branch-and-bound with column generation is called *branch-and-price* (Hooker and van Hoeve, 2018).

Heuristics

A *heuristic* is defined as an "*intuitively designed procedure that does not guarantee an optimal solution*" (Hillier and Lieberman, 2015). As such, problem specific knowledge is exploited in order to define a set of rules that need to be followed in order to construct a good feasible solution. In academic literature, heuristics are often divided in *constructive heuristics* and *improvement heuristics* (Bergh et al., 2013; Ernst et al., 2004a). *Meta-heuristics* and *hyper-heuristics* can be considered special types of heuristics that combine different basic heuristics in order to explore the search space more efficiently (Blum and Roli, 2001; Burke and Kendall, 2005).

Constructive heuristic

A constructive heuristic is a heuristic method that is used to produce a solution from scratch. Since constructive heuristics are considered to be the fastest methods to construct near-optimal feasible solutions from scratch (Abobaker et al., 2011), they are often used for very large and complex problems such as the airline crew rostering problem (Ernst et al., 2004a; Bergh et al., 2013; Blum and Roli, 2001). Constructive heuristic methods include constructing rosters day-by-day, pilot-by-pilot or by assigning high priority activities first (Gamache and Soumis, 1998).

An example of a constructive heuristic is provided in Caprara et al. (1998). The constructive heuristic of Caprara et al. (1998) constructs rosters sequentially for each crew member. For each roster, an initial activity is selected and afterwards, sequential activities are chosen to be added to the roster. Once a roster is completed, the activities in the roster are eliminated from the problem and the next roster will be constructed. In these kinds of constructive heuristics, the challenge is to develop sophisticated algorithms that govern the selection of the consecutive activities to be assigned to the roster.

The main advantages of constructive heuristics are considered their computational efficiency and the transparency in how the rosters are constructed. However, the quality of the solutions produced are highly dependent on the quality of the construction rules where the algorithm is based on. Although a sophisticated constructive heuristic could produce very good or even optimal solutions, a constructive heuristic lacks a global view on the problem and as such the gap between the constructed solution and the optimal solution cannot be quantified (Cappanera and Gallo, 2004).

Improvement heuristics

Improvement heuristics are intuitively designed procedures to improve a solution. An improvement heuristic requires to have an initial solution as input. A common solution method therefore is to first use a constructive heuristic to quickly produce a good initial solution and then exploit improvement heuristic to further improve this initial solution. Improvement heuristics are often iterative procedures, where each iteration tries to further improve the solution.

There are various different kinds of improvement heuristics. *Local search procedures* try to improve the current solution by searching in the neighborhood of the current solution (Hillier and Lieberman, 2015). A local search heuristic is therefore also called a *hill-climbing procedure* (Hillier and Lieberman, 2015). An example of such a local search heuristic is the *gradient search procedure* which aims to find the direction in the solution space in which the solution could be best improved. The gradient search procedure is also referred to as the *steepest descent* or *steepest ascent* method, depending on whether the problem is a minimization or maximization problem. A major drawback of local search methods is that they tend to get stuck in local optima (Hillier and Lieberman, 2015).

For assignment problems such as the airline crew rostering problem, an intuitive improvement procedure is to swap activities of crew members. Jones (1989) for example explores swapping opportunities in order to improve the initially constructed bidlines. The swapping algorithm of Jones (1989) identifies swapping opportunities based on two parameters: the difference in starting date and the difference in pairing length of two candidate pairings. Another swapping procedure proposed by Bianco et al. (1992) identifies "bottlenecks", i.e. assignments with high associated costs, and then seeks a "decreasing alternating chain" to reduce the overall cost of the schedule. This decreasing alternating chain is a rearrangement of assignments among a number of crew members in order to circumvent the bottleneck arc, i.e. an assignment or chain of assignments that induces high costs.

Meta-heuristics

A *meta-heuristic* is defined as a "general solution method that provides both a general structure and strategy guidelines for developing a specific heuristic method to fit a particular kind of problem" (Hillier and Lieberman, 2015). A meta-heuristic algorithm is characterized by its ability to escape from local optima and as such provides a more robust search method than local search heuristics (Hillier and Lieberman, 2015). In academic literature, airline crew rostering models based on the following meta-heuristics have been considered (Lucic and Teodorovic, 2007; Maenhout and Vanhoucke, 2010):

- Simulated Annealing
- Genetic Algorithm
- Tabu Search
- Scatter Search

Lucic and Teodorovic (2007) investigated the use of Simulated Annealing, the Genetic Algorithm and Tabu Search for the airline crew rostering problem, and concluded that the Simulated Annealing algorithm produced the best results. Maenhout and Vanhoucke (2010) proposed a model based on the Scatter Search meta-heuristic, which showed to be capable of producing near-optimal results in fractions of the computational time required by models based on mathematical programming techniques. However, it should be noted that each meta-heuristic has its own characteristics and advantages. Therefore, hybrid methods could for example exploit the global search characteristics of the Genetic Algorithm and the local search capabilities of Tabu Search (Hillier and Lieberman, 2015).

Hyper-heuristics

Hyper-heuristics operate at a higher level of abstraction than (meta-)heuristics and manage which lower level heuristics should be applied at any given time, depending upon the characteristics of the region of the solution space currently under exploration (Cowling et al., 2001). As such, hyper-heuristics can be seen as "*heuristics to choose heuristics*" (Ross, 2005).

There has been no research into the application of hyper-heuristics to the airline crew rostering problem. However, hyper-heuristics have been used to model and solve personnel scheduling problems in other industries. Cowling et al. (2002) and Smet et al. (2014) adopt a hyper-heuristic method to tackle the nurse rostering problem. Misir et al. (2011) take a hyper-heuristic approach to the security personnel routing and rostering problem. In these scheduling problems. the low-level heuristics are for example heuristics that govern the change, swap, addition and dropping of shift assignments (Cowling et al., 2002). The hyper-heuristic as developed by Smet et al. (2014) then employs a "tournament strategy", where these low-level heuristics are competing for the next solution step and the best performing procedure is selected.

Due to the generic nature of a hyper-heuristic algorithm, it enables the modelling of problems with variable characteristics (Cowling et al., 2001; Smet et al., 2014). Also, hyper-heuristics are considered to be cheaper to implement and easier to use than problem specific (meta-)heuristics (Burke et al., 2003; Cowling et al., 2001). In terms of performance, hyper-heuristics are unlikely to outperform exact or well-developed problem-specific methods (Ross, 2005). However, when it is unknown what specific heuristic could be used best for a certain problem, developing a hyper-heuristic could be a good alternative (Ross, 2005).

Artificial Intelligence

Artificial Intelligence (AI) methods solve problems by simulating certain human intelligence processes. Examples of these intelligent processes are *learning*, *reasoning* and *self-correction* (Ernst et al., 2004a). In this section some artificial intelligence techniques will be discussed that can be applied to the personnel rostering problem.

Expert systems

An expert system is aimed to mimic the decision process of a human expert (Ernst et al., 2004a; Jones, 1989). In literature, expert system are also referred to as '*knowledge-based systems*' (Chow and Hui, 1993; Kwok et al., 1995). Also, '*case-based reasoning* (CBR)' is considered closely related. In case-based reasoning, the system stores observed expert decisions on how to repair certain violations in the solution as provided by the case-based reasoning program. This stored knowledge on expert decisions can then be exploited by the program whenever a similar violation is encountered again (Burke et al., 2008).

Problem-specific heuristics, both constructive and improvement heuristics, can be considered expert systems if the rules exploited by the heuristics are developed using expert knowledge. However, expert systems are often interactive and provide decision support rather than being completely automated (Burke et al., 2004; Jones, 1989). Advantages of expert systems over operational research (OR) techniques are considered to be their efficiency in dealing with constraints and their expressiveness in providing explanations when infeasibility is encountered (Kwok et al., 1995). However, the development of an expert system is an intricate process and heavily depends on the availability and quality of expert knowledge (Kwok et al., 1995).

Constraint logic programming

Constraint logic programming (CLP) methods describe complex combinatorial problems using 'domain variables', i.e. variables with a prescribed set of potentially feasible values (Ernst et al., 2004a). CLP methods are exact methods and as such guarantee to provide an optimal solution to optimization problems (Bergh et al., 2013). CLP methods are considered to be efficient techniques to reduce the domain of large combinatorial problems and as such decrease the solution time of the algorithm (Sellmann et al., 2002).

Fahle et al. (2002) exploit CLP for the crew rostering problem in the column generation subproblems in a set-partitioning problem. By doing so, the resource-constrained shortest path problem can be modelled as a constraint and efficiently solved through domain reduction (Fahle et al., 2002). By reducing the domain in the column generation subproblems, the unnecessary generation of numerous infeasible rosters is avoided and the computational efficiency is increased significantly (Sellmann et al., 2002). However, while CLP is considered to be very efficient in finding feasible solutions, CLP is considered less suitable for optimization problems where there are many feasible solutions (Ernst et al., 2004b). As such, CLP is considered particularly useful for highly constrained problems where finding any feasible solution is the main goal (Ernst et al., 2004a).

Machine learning

The concept of 'machine learning' refers to the ability of a computer program to improve its performance based previously generated results (Burke and Kendall, 2005). Three types of machine learning can be distinguished: *supervised*, *unsupervised* and *reinforcement* learning (Burke and Kendall, 2005). With supervised learning an algorithm is trained on a set of input-output pairs, i.e. the algorithm learns to formulate a function that best describes the relations between the input-output pairs. In unsupervised learning the output is not specified, but the algorithm 'learns' to discover statistical patterns within the data set in order to classify the data set. In reinforcement learning, in an iterative process the algorithm receives feedback on the quality of its output where after it adjusts accordingly. Below, the tasks for which machine learning algorithms are mainly applied for are listed (Burke and Kendall, 2005):

- **Classification** - assigning an input to a particular class;
- **Regression** - inferring a function in order to predict values for future input data;
- **Sequence of actions** - finding strategies to choose optimal actions based on the problem state;
- **Data mining** - discovering regularities and patterns in input data.

In the context of airline crew rostering, machine learning could involve learning to improve the rostering process based on an assessment of previously generated rosters. In academic literature no model has been found that incorporates such a learning process. However, Suraweera et al. (2013) developed a model that used machine learning to infer crew rostering constraints from historical crew rosters.

In the broader context of personnel scheduling, two case studies have been found in academic literature. Hao et al. (2004) developed a *neural network* for solving an airport ground staff scheduling problem. Neural networks simulate the human brain by connecting various 'processing units' that are connected with each other (Burke and Kendall, 2005). Each connection between two processing units has a weight factor that represents the relation between the two units. The model of Hao et al. (2004) contains 'assignment units', 'day-off units', 'demand units, 'validation units' and 'work-time units'. The neural network then is aimed to learn the values of the weight factors between these processing units that yield the best solutions. Hao et al. (2004) compare the performance of the developed neural network to three meta-heuristics algorithms: simulated annealing, tabu search and the genetic algorithm. Simulated annealing has been shown to outperform the neural network, both in computational time and solution quality. However, the neural network yields better results than the two other meta-heuristics, again both in computational time and solution quality.

The second case study exploits *pattern recognition* to classify the quality of nurse rosters. The goal of the model of Li et al. (2012) is to exploit pattern recognition to classify the quality of roster solutions on the structure of the rosters rather than on the objective function cost value. The pattern recognition is performed by a neural network where each neuron corresponds to a 'day-employee' slot in the schedule. Each neuron then can have five values, corresponding to an early shift, a day shift, a late shift, a night shift or a day-off. The neural network then is aimed to classify input rosters as being either 'good' or 'bad' rosters, dependent on whether the objective function value is lower or higher than a threshold value. The model has shown to classify rosters correctly in approximately 85% of the cases. The model of Li et al. (2012) can be considered an alternative method to evaluate the quality of rosters without having to calculate the exact value of the objective function. This evaluation model is problem-independent and can be applied to rosters constructed using different approaches and solution methods.

6. Synthesis

This chapter synthesizes the knowledge obtained from the literature study. First, an overview of the academic literature on airline crew rostering and the identified research gaps will be presented. Afterwards, the main insights from the benchmark on crew rostering in other industries will be discussed. Finally, an overview of the relevant solution methods and their characteristics will be presented.

6.1 Airline Crew Rostering

In academic literature on airline crew rostering, a distinction is made between three general approaches: *bidlines*, *equitability-based personalized rostering* and the *Preferential Bidding System (PBS)*. Equitaility-based personalized rostering can be further divided in approaches that include preferences and approaches that do not. Similarly, two types of Preferential Bidding Systems have been encountered in academic literature. With *weighted bids* a priority factor is assigned to the preferences of crew members according to their seniority level. In approaches that adopt *strict seniority* restrictions, the satisfaction of preferences of a senior crew member is always prioritized over the satisfaction of one or more preferences of more junior crew members. A taxonomy of the different airline crew rostering approaches can be seen in Figure 6.1.

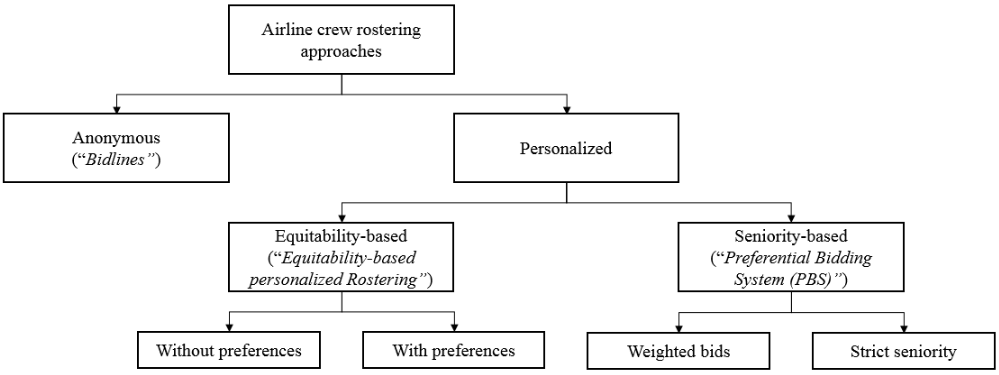

Figure 6.1: Taxonomy of airline crew rostering approaches

Bidlines

The bidlines approach is the approach that used to be common practice in North American airlines (Kasirzadeh et al., 2017). The advantage of bidlines over personalized approaches is that crew members bid for entire rosters. As a result, crew members know exactly how their roster will look like (Kohl and Karisch, 2004). However, after assignment of bidlines, certain pairings conflict with the individual pre-assigned activities of crew members. Those conflicting pairings are then removed from these bidlines and assigned to reserve crew (Campbell et al., 1997). This process is considered less efficient in terms of crew costs than the direct construction of crew rosters as adopted in the personalized approaches (Kohl and Karisch, 2004).

Academic literature on the bidline generation problem focuses on three research efforts: making the bidline generation computationally more efficient such that larger problems can be solved in a shorter amount of time, incorporating quality-of-life considerations within individual bidlines and ensuring an equitable qualtiy-of-life distribution among the generated bidlines. Table 6.1 gives an overview of the bidline models discussed in this literature study.

Table 6.1: Overview of airline crew rostering literature on the bidlines approach (SPP = set-partitioning problem, ES = expert system, ILP = integer linear programming, MCNF = multi-commodity network flow)

Paper	**Obj. Function**	**Model Formulation**	**Solution Method**	**Roster Horizon**	**Indication of run time**
Marsten et al. (1979)	Equitability	SPP	Dual simplex method, branch-and-bound	Week	120 bidlines in 20 hours
Jones (1989)	Costs & Equitability	ES	Constructive heuristic, improvement heuristic	Month	60 bidlines in 1 hour
Jarrah and Diamond (1997)	Costs & Quality	SPP	Column generation, greedy heursitic, improvement heuristic	Month	220 bidlines in 10 minutes
Campbell et al. (1997)	Costs & Quality	ILP	Simulated annealing, greedy heuristic	Month	220 bidlines in 10 hours
Christou et al. (1999)	Costs & Quality	MCNF	Constructive heuristic, genetic algorithm	Month	310 bidlines in 4.5 hours
Weir and Johnson (2004)	Costs & Quality	ILP	Branch-and-price interior point method	Month	150 bidlines in 2 hours
Boubaker et al. (2010)	Equitability	SPP	Branch-and-price, DCA heuristic	Month	560 bidlines in 1 hour
Saddoune et al. (2010)	Costs	SPP	Branch-and-price, DCA heuristic	Month	7500 flights in 44 hours*

* Problem size is in number of flights since the model is an integrated bidline scheduling model

Equitability-based personalized rostering

Outside of North America, equitability-based personalized rostering is the most common approach to the airline crew rostering problem (Kasirzadeh et al., 2017). Equitability-based personalized rostering involves constructing equitable rosters for all crew members directly, taking pre-assigned activities into account. Equitability-based personalized rostering. The personalized rostering problem is considered to be much more difficult to solve than the bidlines problem, because of its increased combinatorial complexity and additional constraints (Grosche, 2009; Ryan, 1992). However, taking pre-assigned activities into account during the rostering process avoids conflicting activities and thus is considered to be more efficient in terms of costs (Kohl and Karisch, 2004).

The majority of the research on equitability-based personalized rostering focuses on developing efficient solution methodologies to solve problems of realistic problem sizes in reasonable computational time. The management of crew preferences is another focus in recent research efforts. Crew preferences can be either included in the pre-assignment of activites, as soft constraints in the objective function or as a hard constraint, for example to ensure that a minimum number of preferences are satisfied. Table 6.2 presents some important characteristic of equitability-based personalized rostering models encountered in academic literature. As a note, if the objective function of a model involves the maximization of the satisfaction of crew preferences this is considered to be a 'quality' component.

Table 6.2: Overview of airline crew rostering literature on equitability-based personalized rostering (SPP = set-partitioning problem, ES = expert system, ILP = integer linear programming, MCNF = multi-commodity network flow)

Paper	Obj. Function	Model Formulation	Solution Method	Roster Horizon	Indication of run time
Nicoletti (1975)	Equitability	MCNF	Day-by-day heuristic, out-of-kilter algorithm	Week	100 crew members in 4 minutes
Ryan (1992)	Costs	SPP	Enumeration, filtering techniques	Month	450 crew members in 2 hours
Gamache et al. (1999)	Costs	SPP	Column generation	Month	380 crew members in 45 minutes
Fahle et al. (2002)	Costs	SPP	Column generation, Constraint programming	Month	70 crew members in 50 minutes
Cappanera and Gallo (2004)	Costs	MCNF	Graph theory, CPLEX	Month	50 crew members in 4 hours
Kohl and Karisch (2004)	Costs & Quality	SPP	Column generation, constructive heuristic, improvement heuristic	Month	1600 crew members in 10 hours
Lucic and Teodorovic (2007)	Equitability	ILP	Greedy heuristic, simulated annealing, genetic algorithm, tabu search	Month	50 crew members in SA: 20 minutes GA: 12 minutes TS: 4 minutes
Maenhout and Vanhoucke (2010)	Costs, Quality & Equitability	SPP	Hybrid scatter search meta-heuristic	2 Weeks	100 crew members in 2 minutes
Kasirzadeh (2017)	Costs & Quality	SPP	Column generation	Month	300 crew members in 3 hours
Armas et al. (2017)	Costs & Equitability	SPP	Multi-start randomized heuristic	Month	20 crew members in 1 minute
Doi et al. (2018)	Equitability	SPP	Greedy heuristic, POPMUSIC matheuristic	Month	300 crew members in 20 minutes

Preferential Bidding System

In a Preferential Bidding System (PBS) the satisfaction of preferences of senior crew members is prioritized over the satisfaction of the preferences of junior crew members. In literature, two ways to do so are considered: *weighted bids* and *strict seniority*. With weighted bids, a priority ranking function is used to promote the preference satisfaction of crew members based on their seniority level as expressed in a weight factor. With strict seniority restrictions, the maximization of the bidding score of a crew member can never go at the expense of the bidding score of a more senior crew member.

The academic literature on the Preferential Bidding System (PBS) is limited. A reason could be that the differentiation in seniority between crew members severely complicates the rostering problem. The PBS problem is considered even more complex than the equitability-based personalized rostering problem, since a set-partitioning problem needs to be solved for each crew member in order to check whether a feasible solution can still be obtained for the remaining crew members (Desaulniers et al., 1998). This significantly increases the complexity of the problem. The computationally expensive backtracking process can result in over 100 hours of run time required to solve a PBS problem of just 90 crew members (Achour et al., 2007). In order to avoid this expensive backtracking process, Gamache et al. (2007) have developed a feasibility test for the PBS with strict seniority restrictions. However, their paper only reported results on the computational time required to perform the feasibility test rather than testing the PBS model with the feasibility test included on the actual crew rostering problem.

Table 6.3: Overview of airline crew rostering literature on the Preferential Bidding System (SPP = set-partitioning problem, ES = expert system, ILP = integer linear programming, MCNF = multi-commodity network flow)

Paper	**Obj. Function**	**Model Formulation**	**Solution Method**	**Roster Horizon**	**Indication of run time**
Gamache and Soumis (1998)	Quality	SPP	Greedy heuristic, column generation	2 Weeks	20 crew members in 10 minutes
Gamache et al. (1998)	Quality	SPP	Greedy heuristic, Column generation	Month	100 crew members in 5 hours
Achour et al. (2007)	Quality	SPP	Column generation	Month	*highly variable*
Gamache et al. (2007)	Quality	SPP	Column generation, graph coloring, tabu search	Month	*not reported*

6.2 Research gaps

Academic literature on airline crew rostering has been focused on developing more efficient solution methods within a certain specific crew rostering approach. However, a crew rostering approach should not be considered a fixed and rigid framework. A consistent evaluation of the impact of crew rostering elements on the efficiency and quality of the generated rosters could provide insights into how airlines should formulate their crew rostering approach. The following research gaps concerning these crew rostering elements have been identified:

1. **Integration of the rostering process -** the assignment of pairings, vacations, training activities and reserve blocks could be integrated into one rostering process. It can be investigated whether the integration of the rostering process of these different activities results in increased roster efficiency;
2. **Impact of equitability and seniority measures -** equitability and seniority considerations can be taken into account as constraints or objective function coefficients in the crew rostering model. The impact of these different measures on the cost and the quality of the generated roster can be evaluated;
3. **Simulation of the dynamic rostering process -** the rolling horzion procedure can be simulated. With a dynamic roster model, global constraint consistency, the influx of crew requests and the effect of disturbances can be investigated;
4. **Consistent evaluation of crew rosters -** a method for the consistent evaluation of rosters can be developed. This method should take the generated rosters as input and should be independent of the approach and method adopted in the generation of the rosters.

6.3 Benchmark

A benchmark was performed to get a better understanding of the generic personnel rostering problem and its application in different contexts. Following different classifications, the airline crew rostering problem could be considered a *tour scheduling* problem, a *non-cyclic* scheduling problem and a *mobility centred* planning problem. Also, the airline crew rostering problem could be seen as integrating the *days-off scheduling*, *lines of work construction* and *staff assignment* modules of personnel scheduling.

Also, the literature on personnel rostering in the public transit, health care and call centre industries was assessed. In these other domains of personnel rostering, the dissonance in researchers' opinions on the best solution methodology for the rostering problem was considerably smaller. In the airline industry, researchers consider both mathematical programming methods and (meta-)heuristic methods. In the other industries however, the overall consensus was that large rostering problems are too complex to be solved by mathematical programming methods and that heuristics are needed in order to refrain the model from becoming intractable.

6.4 Solution Methods

The crew rostering problem is a combinatorial optimization problem. Researchers have adopted different ways of representing the problem mathematically and different computational techniques to solve the problem. Table 6.4 provides an overview of the main solution methods encountered in academic literature.

Table 6.4: An overview of the main solution methods for crew rostering problems

Model Formulation	**Network Representation**	**Solution Techniques: Math. Program.**	**Solution Techniques: Heuristics**	**Solution Techniques: AI**
Integer Linear Programming	Multi-commodity network flow	LP relaxation	Constructive heuristics	Expert systems
Set-Partitioning Problem	Time-space network	Lagrangian relaxation	Improvement heuristics	Constraint programming
Stochastic programming		Branch-and-bound	Meta-heuristics	Machine learning
Dynamic modelling		Column generation	Hyper-heuristics	

Mathematical programming methods for the crew rostering problem are exact methods that solve for optimality using branch-and-bound and column generation. Numerous researchers have pointed out that realistic, large crew rostering problems are too complex for mathematical programming methods (Desaulniers and Hickman, 2007; Hojati, 2010; Maenhout and Vanhoucke, 2010; Xie et al., 2017). Especially for research purposes, hours of running times induced by mathematical programming methods are considered inadequate. Heuristics on the other hand exploit problem-specific knowledge which significantly speeds up the rostering process. The most striking example is the model of Maenhout and Vanhoucke (2010). In an average computational time of only 127 seconds the hybrid scatter search heuristic obtains results that deviate an average of 1.30% from the result obtained by the exact branch-and-price algorithm with an average run time of over 20 hours.

Heuristic methods for the crew rostering problem often approach the problem in two phases. First, using a constructive heuristic an initial solution is produced from scratch. Afterwards, an improvement heuristic or meta-heuristic is used to improve this initial solution. In academic literature, no research has been found that used hyper-heuristics or machine learning for the airline crew rostering problem. There have been some research studies using these techniques for other personnel scheduling problems, but also in this broader context the potential of using these novel solution methods for the personnel scheduling problem is yet to be explored. Hyper-heuristics show promising results when the aim is to develop a fast and generic research model. Also, hyper-heuristics allow for the incorporation of machine learning techniques, learning to select the appropriate low-level heuristic from historical data (Burke et al., 2010).

7. Conclusion

The purpose of this literature study was threefold: to provide an overview of the academic literature on airline crew rostering, to identify potential subjects for future studies and to identify modelling techniques that can be used to address the identified research gaps.

Regarding the first objective, it has been found that airline crew rostering can be done in various ways following different approaches. North American airlines traditionally adopt the bidlines approach, where anonymous rosters are made on which the crew members place their bids. In the rest of the world, personalized rostering approaches are more common. Academic literature on crew rostering has followed separated paths for these different approaches, each of which focused on developing efficient solution methods to solve the crew rostering problem within the framework associated with a specific crew rostering approach. However, in reality, CLA rules and regulations evolve over time through negotiations between airlines and labor unions. Besides these CLA regulations, other strategic decisions of airlines concerning the rostering process include how crew preferences are managed and what objectives are pursued. As such, a crew rostering approach should not be considered a fixed and rigid framework. Future studies therefore should provide insights on how certain *elements* of a crew rostering approach affect the efficiency and quality of the rostering process. The following crew rostering elements have been identified:

(a) *Pre-assigned activities* - pre-scheduled activities such as vacations and training days;

(b) *Crew preferences* - pairings or days off preferred by crew members;

(c) *Equitability measures* - measures to create an even balance in roster characteristics;

(d) *Seniority measures* - measures to prioritize the satisfaction of senior crew members;

(e) *Roster horizon* - the number of days or weeks before operation for which a crew roster is generated;

(f) *Roster period* - the number of days or weeks of the period that is rostered.

Four different research gaps have been identified, each of which deals with certain crew rostering elements:

1. **Integration of the rostering process** - the assignment of pairings, vacations, training activities and reserve blocks could be integrated into one rostering process. It can be investigated whether the integration of the rostering process of these different activities results in increased roster efficiency;

2. **Impact of equitability and seniority measures** - equitability and seniority considerations can be taken into account as constraints or objective function coefficients in the crew rostering model. The impact of these different measures on the cost and the quality of the generated roster can be evaluated;

3. **Simulation of the dynamic rostering process** - the rolling horzion procedure can be simulated. With a dynamic roster model, global constraint consistency, the influx of crew requests and the effect of disturbances can be investigated;

4. **Consistent evaluation of crew rosters** - a method for the consistent evaluation of rosters can be developed. This method should take the generated rosters as input and should be independent of the approach and method adopted in the generation of the rosters.

From the academic literature on personnel rostering, an overview was made on what modelling techniques could be used for the assessment of the identified research gaps. The mathematical model formulation of the problem, its network representation and possible solution techniques were discussed. Numerous researchers have pointed out that realistic, large crew rostering problems are too complex for mathematical programming methods. Especially for research purposes, computational speed might be considered more important than proven optimality. Heuristics exploit problem-specific knowledge which can significantly speed up the rostering process. In academic literature on airline crew rostering, most heuristic methods combine constructive heuristic with improvement (meta-)heuristics. In a benchmark on personnel rostering in other industries, two alternative solution methods were found: hyper-heuristics and machine learning. Future studies could test the viability of using these novel methods for the airline crew rostering problem.

The identified research gaps require the simulation and testing of different crew rostering approaches. Adaptivity and computational efficiency are therefore considered essential requirements for the crew rostering model that should be developed. The research objective therefore is the following:

Research objective: *to provide insight into the impact of crew rostering elements on the efficiency and quality of the generated rosters, by developing a fast and adaptive crew rostering model, simulating different rostering approaches and evaluating the generated crew rosters.*

The assessment of the impact of crew rostering elements can provide airlines with insights on how the efficiency of their adopted rostering approach could be improved or how they can improve the quality-of-life aspects of their crew schedules. Moreover, a fast and adaptive crew rostering model could serve as a simulation tool for airlines during Collective Labour Agreement (CLA) negotiations with labour unions. Such a simulation model can quantify the impact of new rules and regulations on the efficiency and quality of the crew rostering process. This is considered a major asset for airlines during CLA negotiations, assisting airlines in efficiently picking their battles.

Bibliography

Abobaker, R., Ayob, M., and Hadwan, M. (2011). Greedy constructive heuristic and local search algorithm for solving Nurse Rostering Problems. In *2011 3rd Conference on Data Mining and Optimization (DMO)*, pages 194–198.

Achour, H., Gamache, M., Soumis, F., and Desaulniers, G. (2007). An Exact Solution Approach for the Preferential Bidding System Problem in the Airline Industry. *Transportation Science*, 41(3):354–365.

Aksin, O., Armony, M., and Mehrotra, V. (2007). The Modern Call Center: A MultiDisciplinary Perspective on Operations Management Research. *Production and Operations Management*, 16(6):665–688.

Armas, J., Cadarso, L., Juan, A., and Faulin, J. (2017). A multi-start randomized heuristic for real-life crew rostering problems in airlines with work-balancing goals. *Annals of Operations Research*, 258(2):825–848.

Asgeirsson, E. (2014). Bridging the gap between self schedules and feasible schedules in staff scheduling. *Annals of Operations Research*, 218(1):51–69.

Asgeirsson, E. and Sigurdardottir, G. (2016). Near-optimal MIP solutions for preference based self-scheduling. *Annals of Operations Research*, 239(1):273–293.

Avramidis, A., Chan, W., and L'Ecuyer, P. (2009). Staffing multi-skill call centers via search methods and a performance approximation. *IIE Transactions*, 41(6):483–497.

Baker, K. (1976). Workforce allocation in cyclical scheduling problems: a survey. *Journal of the Operational Research Society*, 27(1):155–167.

Balas, E. and Padberg, M. (1976). Set Partitioning: A Survey. *SIAM Review*, 18(4):710–760.

Bard, J. and Purnomo, H. (2005). Preference scheduling for nurses using column generation. *European Journal of Operational Research*, 164(2):510–534.

Barnhart, C. and Cohn, A. (2004). Airline Schedule Planning: Accomplishments and Opportunities. *Manufacturing & Service Operations Management*, 6(1):3–22.

Belobaba, P., Odonoi, A., and Barnhart, C. (2009). *The Global Airline Industry*. John Wiley & Sons.

Bergh, van den, J., Belin, J., de Bruecker, P., Demeulemeester, E., and de Boeck, L. (2013). Personnel scheduling: A literature review. *European Journal of Operations Research*, 226(3):367–385.

Bianco, L., Bielli, M., Mingozzi, A., Ricciardelli, S., and Spadoni, M. (1992). A heuristic procedure for the crew rostering problem. *European Journal of Operational Research*, 58(2):272–283.

Blum, C. and Roli, A. (2001). Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Computing Surveys*, 35:268–308.

Boubaker, K., Desaulniers, G., and Elhallaoui, I. (2010). Bidline scheduling with equity by heuristic dynamic constraint aggregation. *Transportation Research Part B: Methodological*, 44(1):50–61.

Bourdais, S., Galinier, P., and Pesant, G. (2003). Hibiscus: A constraint programming application to staff scheduling in health care. In *Principles and Practice of Constraint Programming*, pages 153–167. Springer Berlin Heidelberg.

Brucker, P., Qu, R., and Burke, E. (2011). Personnel scheduling: Models and complexity. *European Journal of Operational Research*, 210(3):467–473.

Burke, E., Curtois, T., Post, G., Qu, R., and Veltman, B. (2008). A hybrid heuristic ordering and variable neighbourhood search for the nurse rostering problem. *European Journal of Operational Research*, 188:3–27.

Burke, E., de Causmaecker, P., vanden Berghe, G., and van Landeghem, H. (2004). The State of the Art in Nurse Rostering. *Journal of Scheduling*, 7:441–499.

Burke, E., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., and Woodward, J. (2010). *A Classification of Hyper-heuristic Approaches*, pages 449–468. Springer US.

Burke, E. and Kendall, G. (2005). *Search Methodologies - Introductory Tutorials in Optimization and Decision Support Techniques*. Springer.

Burke, E., Kendall, G., Newall, J., Hart, E., Ross, P., and Schulenburg, S. (2003). *Hyper-Heuristics: An Emerging Direction in Modern Search Technology*, pages 457–474. Springer US.

Campbell, K., Durfee, R., and Hines, G. (1997). FedEx Generates Bid Lines Using Simulated Annealing. *Interfaces*, 27(2):1–16.

Cappanera, P. and Gallo, G. (2004). A Multicommodity Flow Approach to the Crew Rostering Problem. *Operations Research*, 52(4):583–596.

Caprara, A., Toth, P., Vigo, D., and Fischetti, M. (1998). Modeling and Solving the Crew Rostering Problem. *Operations Research*, 46(6):820–830.

Carraresi, P. and Gallo, G. (1984). A multi-level bottleneck assignment approach to the bus drivers' rostering problem. *European Journal of Operational Research*, 16(2):163–173.

Cezik, M. and L'Ecuyer, P. (2008). Staffing Multiskill Call Centers via Linear Programming and Simulation. *Management Science*, 54(2):310–323.

Cheang, B., Li, H., Lim, A., and Rodrigues, B. (2003). Nurse rostering problems - a bibliographic survey. *European Journal of Operational Research*, 151(3):447–460.

Chow, K. and Hui, C. (1993). Knowledge-based system for rostering. *Expert Systems with Applications*, 6(3):361–375.

Christou, I., Zakarian, A., Liu, J., and Carter, H. (1999). A Two-Phase Genetic Algorithm for Large-Scale Bidline-Generation Problems at Delta Air Lines. *Interfaces*, 29(5):51–65.

Cowling, P., Kendall, G., and Soubeiga, E. (2001). A Hyperheuristic Approach to Scheduling a Sales Summit. In Burke, E. and Erben, W., editors, *Practice and Theory of Automated Timetabling III*, pages 176–190. Springer Berlin Heidelberg.

Cowling, P., Kendall, G., and Soubeiga, E. (2002). Hyperheuristics: A Robust Optimisation Method Applied to Nurse Scheduling. In Guervós, J., Adamidis, P., Beyer, H., Schwefel, H., and Fernández-Villacañas, J., editors, *Parallel Problem Solving from Nature*, pages 851–860. Springer Berlin Heidelberg.

De Causmaecker, P., Demeester, P., Vanden Berghe, G., and Verbeke, B. (2004). Analysis of real-world personnel scheduling problems. *In: Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling*, pages 183–198.

Desaulniers, G., Desrosiers, J., Gamache, M., and Soumis, F. (1998). *Crew Scheduling in Air Transportation*, pages 169–185. Springer US.

Desaulniers, G. and Hickman, M. D. (2007). Chapter 2 Public Transit. In *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, pages 69–127. Elsevier.

Desrosiers, J., Dumas, Y., Solomon, M., and Soumis, F. (1995). Time constrained routing and scheduling. *Handbooks in Operations Research and Management Science*, 8:35–139.

Deveci, M. and Demirel, N. (2018). A survey of the literature on airline crew scheduling. *Engineering Applications of Artificial Intelligence*, 74:54–69.

Doi, T., Nishi, T., and Voß, S. (2018). Two-level decomposition-based matheuristic for airline crew rostering problems with fair working time. *European Journal of Operational Research*, 267(2):428–438.

Elhallaoui, I., Villeneuve, D., Soumis, F., and Desaulniers, G. (2005). Dynamic Aggregation of Set-Partitioning Constraints in Column Generation. *Operations Research*, 53(4):632–645.

Eltoukhy, E., Chan, F., and Chung, S. (2017). Airline schedule planning: a review and future directions. *Industrial Management & Data Systems*, 117(6):1201–1243.

Ernst, A., Jiang, H., Krishnamoorthy, M., Owens, B., and Sier, D. (2004a). An Annotated Bibliography of Personnel Scheduling and Rostering. *Annals of Operations Research*, 127(1-4):21–144.

Ernst, A., Jiang, H., Krishnamoorthy, M., and Sier, D. (2004b). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*, 153(1):3–27.

Fahle, T., Junker, U., Karisch, S., Kohl, N., Sellmann, M., and Vaaben, B. (2002). Constraint Programming Based Column Generation for Crew Assignment. *Journal of Heuristics*, 8(1):59–81.

Ford, L. and Fulkerson, D. (1962). *Flows in Networks*. Princeton University Press.

Fukunaga, A., Hamilton, E., Fama, J., Andre, D., Matan, O., and Nourbakhsh, I. (2002). Staff scheduling for inbound call centers and customer contact centers. *Eighteenth National Conference on Artificial Intelligence*, pages 822–829.

Gamache, M., Hertz, A., and Ouellet, J. (2007). A graph coloring model for a feasibility problem in crew scheduling. *Computers & Operations Research*, 34(8):2384–2395.

Gamache, M. and Soumis, F. (1998). A method for optimally solving the rostering problem. *Operations Research in the Airline Industry*, 9:124–157.

Gamache, M., Soumis, F., and Marquis, G. (1999). A Column Generation Approach for Large-Scale Aircrew Rostering Problems. *Operations Research*, 47(2):247–263.

Gamache, M., Soumis, F., Villeneuve, D., Desrosiers, J., and Glinas, E. (1998). The Preferential Bidding System at Air Canada. *Transportation Science*, 32(3):246–255.

Gans, N., Koole, G., and Mandelbaum, A. (2003). Telephone Call Centers: Tutorial, Review, and Research Prospects. *Manufacturing Service Operations Management*, 5(2):79–141.

Gopalakrishnan, B. and Johnson, E. (2005). Airline Crew Scheduling: State-of-the-Art. *Annals of Operations Research*, 140(1):305–337.

Grosche, T. (2009). *Computational Intelligence in Integrated Airline Scheduling*. Springer-Verlag.

Hansen, P. and Mladenovic, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3):449–467.

Hao, G., Lai, K., and Tan, M. (2004). A Neural Network Application in Personnel Scheduling. *Annals of Operations Research*, 128:65–90.

Hillier, F. and Lieberman, G. (2015). *Introduction to Operations Research*. McGraw-Hill Education.

Hojati, M. (2010). Near-optimal solution to an employee assignment problem with seniority. *Annals of Operations Research*, 181(1):539–557.

Hooker, J. N. and van Hoeve, W. (2018). Constraint programming and operations research. *Constraints*, 23(2):172–195.

Ingels, J. and Maenhout, B. (2018). The impact of overtime as a time-based proactive scheduling and reactive allocation strategy on the robustness of a personnel shift roster. *Journal of Scheduling*, 21(2):143–165.

Ionescu, L. and Kliewer, N. (2011). Increasing Flexibility of Airline Crew Schedules. *Procedia - Social and Behavioral Sciences*, 20:1019–1028.

Irnich, S. and Desaulniers, G. (2005). *Shortest Path Problems with Resource Constraints*, pages 33–65. Springer US.

Jarrah, A. and Diamond, J. (1997). The Problem of Generating Crew Bidlines. *Interfaces*, 27(4):1–90.

Jones, R. (1989). Development of an Automated Airline Crew Bid Generation System. *Interfaces*, 19(4):44–51.

Kasirzadeh, A., Saddoune, M., and Soumis, F. (2017). Airline crew scheduling: models, algorithms, and data sets. *EURO Journal on Transportation and Logistics*, 6(2):111–137.

Kohl, N. and Karisch, S. (2004). Airline Crew Rostering: Problem Types, Modeling, and Optimization. *Annals of Operations Research*, 127(1-4):223–257.

Kwok, L., Hung, S., and Pun, C. (1995). Knowledge-based cabin crew pattern generator. *Knowledge-Based Systems*, 8(1):55–63.

Li, J., Burke, E., and Qu, R. (2012). A pattern recognition based intelligent search method and two assignment problem case studies. *Applied Intelligence*, 36(2):442–453.

Lucic, P. and Teodorovic, D. (2007). Metaheuristic approach to the aircrew rostering problem. *Annals of Operations Research*, 155(1):311–338.

Maenhout, B. and Vanhoucke, M. (2010). A hybrid scatter search heuristic for personalized crew rostering in the airline industry. *European Journal of Operational Reserach*, 206(1):155–167.

Marsten, R., Muller, M., and Killion, C. (1979). Crew Planning at Flying Tiger: A Successful Application of Integer Programming. *Management Science*, 25(12):1175–1183.

Marsten, R. and Shepardson, F. (1981). Exact Solution of Crew Scheduling Problems Using the Set Partitioning Model: Recent Successful Applications. *Networks*.

Mihaylov, M., Smet, P., Van Den Noortgate, W., and Vanden Berghe, G. (2016). Facilitating the transition from manual to automated nurse rostering. *Health Systems*, 5(2):120–131.

Millar, H. and Kiragu, M. (1998). Cyclic and non-cyclic scheduling of 12 h shift nurses by network programming. *European Journal of Operational Research*, 104(3):582–592.

Misir, M., Smet, P., Verbeeck, K., and Vanden Berghe, G. (2011). Security personnel routing and rostering: a hyper-heuristic approach. volume 3, pages 193–205.

Morris, J. and Showalter, M. (1983). Simple Approaches to Shift, Days-Off and Tour Scheduling Problems. *Management Science*, 29(8):869–986.

Musliu, N., Gartner, J., and Slany, W. (2002). Efficient generation of rotating workforce schedules. *Discrete Applied Mathematics*, 118(1-2):85–98.

Nicoletti, B. (1975). Automatic Crew Rostering. *Transportation Science*, 9(1):1–97.

Odoni, A., Rousseau, J., and Wilson, N. (1994). Chapter 5 Models in urban and air transportation. In *Operations Research and The Public Sector*, volume 6 of *Handbooks in Operations Research and Management Science*, pages 107–150. Elsevier.

Rocha, M., Oliveira, J., and Carravilla, M. (2013). Cyclic staff scheduling: optimization models for some real-life problems. *Journal of Scheduling*, 16(2):231–242.

Rosenberger, J., Schaefer, A., Goldsman, D., Johnson, E., Kleywegt, A., and Nemhauser, G. (2002). A Stochastic Model of Airline Operations. *Transportation Science*, 36(4):357–377.

Ross, P. (2005). *Hyper-Heuristics*, pages 529–556. Springer US.

Ryan, D. (1992). The solution of massive generalized set partitioning problems in aircrew rostering. *Journal of the Operational Research Society*, 43(5):459467.

Saddoune, M., Desaulniers, G., Elhallaoui, I., and Soumis, F. (2012). Integrated Airline Crew Pairing and Crew Assignment by Dynamic Constraint Aggregation. *Transportation Science*, 46(1):39–55.

Salassa, F. and Vanden Berghe, G. (2012). A stepping horizon view on nurse rostering. *Proceedings of the 9the International Conference on the Practice and Theory of Automated Timetabling*, pages 161–174.

Santos, H., Toffolo, T., Gomes, R., and Ribas, S. (2016). Integer programming techniques for the nurse rostering problem. *Annals of Operations Research*, 239(1):225–251.

Sellmann, M., Zervoudakis, K., Stamatopoulos, P., and Fahle, T. (2002). Crew Assignment via Constraint Programming: Integrating Column Generation and Heuristic Tree Search. *Annals of Operations Research*, 115(1):207–225.

Silvestro, R. and Silvestro, C. (2000). An evaluation of nurse rostering practices in the National Health Service. *Journal of Advanced Nursing*, 32(3):525–535.

Smet, P., Bilgin, B., De Causmaecker, P., and Vanden Berghe, G. (2014). Modelling and evaluation issues in nurse rostering. *Annals of Operations Research*, 218(1):303–326.

Smet, P., Salassa, F., and Vanden Berghe, G. (2016). Local and global constraint consistency in personnel rostering. *International Transactions in Operational Research*, 24(5):1099–1117.

Suraweera, P., Webb, G., Evans, I., and Wallace, M. (2013). Learning crew scheduling constraints from historical schedules. *Transportation Research Part C: Emerging Technologies*, 26:214–232.

Tien, J. and Kamiyama, A. (1982). On Manpower Scheduling Algorithms. *SIAM Review*, 24(3):275–287.

Valouxis, C., Gogos, C., Goulas, G., Alefragis, P., and Housos, E. (2012). A systematic two phase approach for the nurse rostering problem. *European Journal of Operational Research*, 219(2):425–433.

Warner, D. (1976). Scheduling Nursing Personnel According to Nursing Preference: A Mathematical Programming Approach. *Operations Research*, 24(5):842–856.

Weir, J. and Johnson, E. (2004). A Three-Phase Approach to Solving the Bidline Problem. *Annals of Operations Research*, 127(1-4):283–308.

Xie, L., Merschformann, M., Kliewer, N., and Suhl, L. (2017). Metaheuristics approach for solving personalized crew rostering problem in public bus transit. *Journal of Heuristics*, 23(5):321–347.

Xie, L. and Suhl, L. (2014). Cyclic and non-cyclic crew rostering problems in public bus transit. *Operations Research Spectrum*, 37(1):99–136.

Part III

Appendices

(Not graded yet)

APPENDIX A: ALGORITHM SELECTION

Ten different algorithms have been tested for the pairing assignment module of the simulation model. This section discusses the algorithms, their compared performance and finally the selection of the algorithm adopted in the simulation model.

A. Algorithms

Below, a short explanation of each of the ten tested pairing assignment algorithms can be found.

1) Random: The random algorithm is a constructive algorithm where at every step a random pairing is selected. Pilots are then filtered based on whether the selected pairing fits in the roster of the pilot, after which the pairing is assigned to a random available pilot.

2) Heuristic 1: Based on the assignment relaxation of the mathematical model, the first of the two constructive heuristics first selects a pilot and then selects the pairing that should be assigned to that pilot. The pilot selection heuristic selects the pilot with the highest cost if no pairing would be assigned to that pilot, i.e. either cost due to lost days or due to a penalty for next week's roster. If multiple pilots have the same cost if no pairing is assigned, the pilot with the minimum number of assignment options is selected. After the pilot is selected, the pairing heuristic first filters on those pairings that can be assigned to the selected pilot. The pairing with the minimum assignment cost then is selected to be assigned to the selected pilot. If multiple pairings have the same assignment cost, the pairing with the minimum assignment options is selected.

3) Heuristic 2: Based on the assignment relaxation of the mathematical model, the second constructive heuristic first selects a pairing and then selects a pilot that will get that pairing assigned. The pairing with the minimum number of assignment options is selected. After the pairing is selected, the pilot heuristic first filters on those pilots that can have the selected pairing assigned. Afterwards, the pilot that has the minimum number of assignment options is selected.

4) LP-bipartite: The Linear Programming (LP) bipartite algorithm is based on the bipartite relaxation of the mathematical model. The LP-bipartite algorithm exploits the Gurobi Optimizer in order to find the minimum cost bipartite solution, i.e. the minimum cost solution with each pilot having at most one pairing assigned.

5) LP-complete: The Linear Programming (LP) complete algorithm exploits the Gurobi Optimizer in order to find the minimum cost solution to the complete set-partitioning problem. The solution of the LP-complete algorithm is considered the optimal solution and serves as the reference for solution quality for the other algorithms.

6) Hungarian: This pairing assignment algorithm uses the Hungarian algorithm in order to find the minimum cost bipartite solution. The Hungarian algorithm is a classic assignment algorithm that solves the weighted bipartite matching problem by performing a set of matrix operations on the bipartite cost matrix that encompasses the roster cost of each pilot-pairing assignment (Hillier & Lieberman, 2015).

7) Hungarian Heuristic: The Hungarian Heuristic algorithm is a constructive algorithm where the Hungarian algorithm is applied iteratively. The complete set of pairings is split up in two sets, split up by starting date of the pairing. The split is made such that each set has no two pairings that could potentially be assigned to a single pilot. Afterwards, the problem is solved by first assigning the first set of pairings using the Hungarian bipartite algorithm and then doing the same for the second set of pairings.

8) Hungarian Improvement All: The Hungarian Improvement algorithm can be considered an improvement heuristic with the Hungarian bipartite solution as the initial solution. This initial solution then is improved by searching for assignments that improve the remaining bipartite solution. As such, the algorithm exploits the resemblance of the weekly crew rostering problem to a minimum-cost bipartite matching problem.

9) Hungarian Improvement 5: This algorithm follows the same method as the Hungarian Improvement All algorithm. The difference is that this configuration considers just five pairings per pilot in the search for assignments that would improve the remaining bipartite solution.

10) Hungarian Improvement 1: This algorithm follows the same method as the Hungarian Improvement All algorithm. The difference is that this configuration considers just one pairing per pilot in the search for assignments that would improve the remaining bipartite solution.

B. Algorithm Comparison

The algorithms have been tested on 44 problem instances that correspond to different roster weeks of a small long-haul aircraft division. The experiments have been conducted on a HP Elitebook 8570w with an Intel(R) Core(TM) processor running at a rate of 2.40 GHz. The algorithms are coded in Python. For the Linear Programming (LP) algorithms, version 8.1 of the Gurobi Optimizer is used.

The algorithms have been compared on three performance criteria: solution quality, computational time and adaptability.

1) Solution Quality: The LP-complete algorithm finds the minimum cost solution to the complete problem and serves as a reference for algorithm comparison on solution quality. The median values of the different algorithms have been compared to the median optimal solution as produced by the LP-complete algorithm. In the evaluation, per algorithm the deviation of the median from optimal is averaged over the two problem sets. The results, depicted in box plots, can be found in Figures 17 and 18.

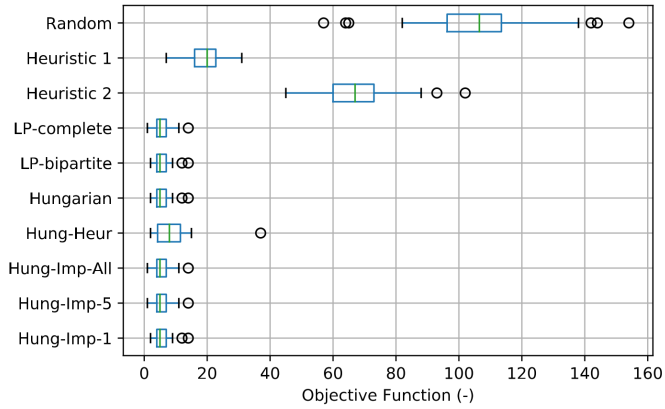

Fig. 17. Algorithm Comparison: Objective Function (1)

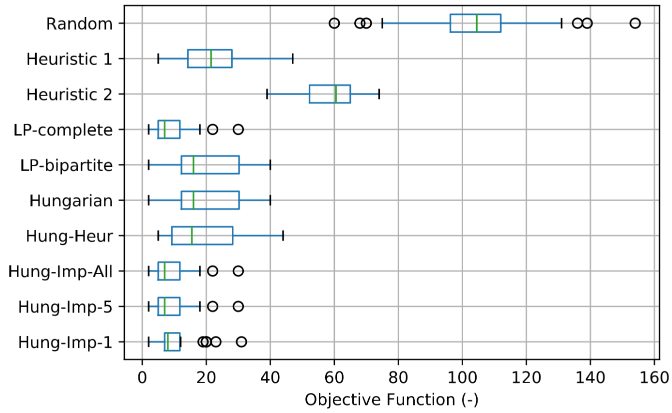

Fig. 18. Algorithm Comparison: Objective Function (2)

2) Computational Time: The added value of the developed simulation model over commercial software is its adaptability and speed. Therefore, besides solution quality, computational time was the second major evaluation criteria during algorithm selection. Figures 19 and 20 present the box plot comparison in computational time of the different algorithms. Important to state is that the computational time as reported in this subsection and on which the algorithms are compared is the computational time required solely for pairing assignment. Pre-processing and post-processing, which takes around 90 seconds these problem instances, is not included in the algorithm evaluation and comparison.

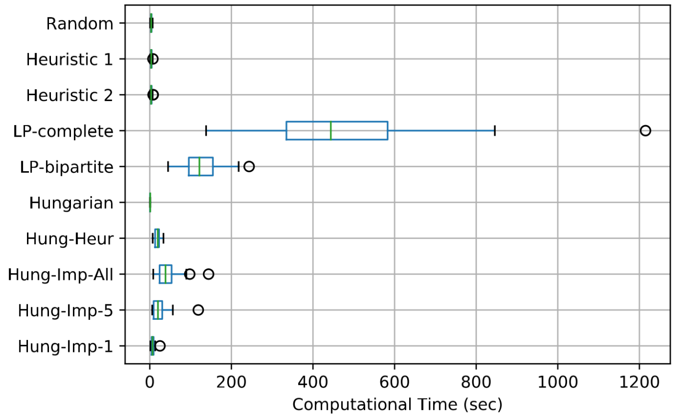

Fig. 19. Algorithm Comparison: Computational Time (1)

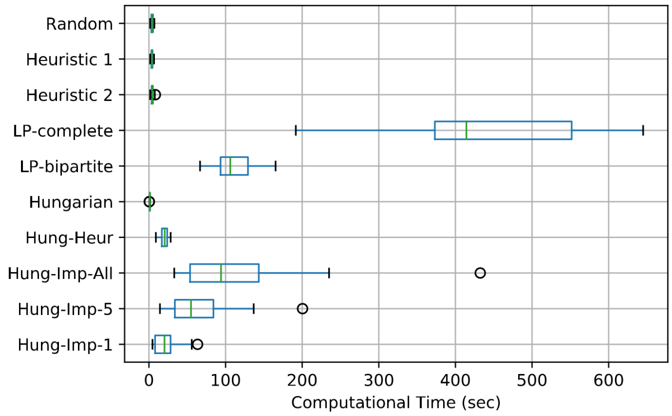

Fig. 20. Algorithm Comparison: Computational Time (2)

3) Adaptability: Besides having high solution quality and low computational time, an additional requirement for the pairing assignment algorithm in the simulation model is that it should be '*adaptable*'. In this study, the adaptability of the algorithms is assessed on two aspects. Firstly, the algorithm should be capable of handling changing rostering rules. Secondly, the algorithm should perform well for varying problem sizes.

The first aspect of these two is assessed qualitatively. Overall, the algorithms that exploit problem-specific knowledge to make pilot-pairing assignment decisions are expected to have difficulties in adapting to changing rostering rules. The algorithms that fall into this category are the constructive heuristics, Heuristic 1 and Heuristic 2. These algorithms therefore are considered to be less adaptable than the algorithms that rely solely on a pilot-pairing cost matrix, the construction of which is easily adaptable to changing rostering rules.

Regarding the second aspect, the adaptability to varying problem sizes, the pairing assignment algorithms have been tested on a number of problem instances that correspond to roster weeks of a medium size long-haul aircraft division. These problem involve a significant increase in problem size, having around 450 pilots and 270 pairings compared to around 150 pilots and 100 pairings for the small long-haul aircraft division used for the other evaluation criteria. Considering that the simulation model should also be able to solve these larger problems efficiently, the computational time required by the algorithms to solve these large problems has been tested.

From the results as presented in Table II it becomes clear that the computational time of the linear programming algorithms and specifically the LP-complete algorithm greatly increases for larger rostering problems. The constructive heuristics and the Hungarian algorithm on the other hand seem to be barely affected by the increase in problem size. The Hungarian Improvement algorithm is considered particularly adaptable to variations in problem sizes, since its configuration can be adjusted to the size of the considered problem instance. For large problems, the search for assignments that improve the remaining bipartite solution can be limited in order to gain computational efficiency at a small cost of solution quality.

TABLE II

ALGORITHM COMPARISON: ADAPTABILITY

	Problem Instance 1	Problem Instance 2
Random	27s	30s
Heuristic 1	27s	25s
Heuristic 2	33s	31s
LP-complete	8h 03m	9h 19m
LP-bipartite	1h 06m	1h 26m
Hungarian	5s	5s
Hung-Heur	3m 08s	3m 28s
Hung-Imp-All	31m 21s	22m 09s
Hung-Imp-5	16m 32s	12m 04s
Hung-Imp-1	4m 52s	3m 34s

C. *Algorithm Selection*

The algorithms are compared on three performance criteria: solution quality, computational time and adaptability. All three criteria are considered equally important and therefore the total score of an algorithm in the trade-off is the sum of the scores on each individual criterion. A five-point grading scale is adopted, ranging from -2 ($--$) to +2 ($++$).

Table III presents the compared performance of the different algorithms. As illustrated in the table, the random algorithm and the constructive heuristics are considered inferior because of their low solution quality. Moreover, the constructive heuristics have a low score on adaptability due to the expected difficulties in adjusting the algorithm to changing rostering rules. The Linear Programming (LP) algorithms on the other hand are discarded for their inferior computational efficiency. Also, the computational time required by the LP algorithms is presumed to aggravate as problems increase in size, resulting in a low score on algorithm adaptability.

The Hungarian-based algorithms have been found to outperform the other algorithms. In particular the Hungarian Improvement algorithm is considered superior to the other algorithms for the specific problem as formulated in this research. The Hungarian Improvement algorithm succeeds in providing rosters of near-optimal solution quality, within acceptable computational time. Moreover, the Hungarian Improvement algorithm is considered particularly adaptable, as it is both adaptable to changing rostering rules and as it allows for the adjustment of its configuration in order to adapt to problems of varying sizes. Therefore, the Hungarian Improvement algorithm has been selected as the pairing assignment algorithm in the simulation model.

TABLE III

ALGORITHM SELECTION: TRADE-OFF TABLE

	Solution Quality	Computational Time	Adaptability	Total
Random	- -	++	+	+
Heuristic 1	-	++	-	+/-
Heuristic 2	- -	++	-	-
LP-complete	++	- -	- -	- -
LP-bipartite	+/-	-	-	- -
Hungarian	+/-	++	+	+++
Hung-Heur	+/-	+	+	++
Hung-Imp-All	++	-	++	+++
Hung-Imp-5	++	+/-	++	++++
Hung-Imp-1	+	+	++	++++

APPENDIX B: MODEL VALIDATION

The developed simulation model is validated using actual historical rosters from the collaborating airline. For model validation, the 'LP-complete' algorithm is used, since it is the Linear Programming (LP) algorithm that solves the crew rostering problem optimally in terms of the objective function adopted in the simulation model. The LP-complete algorithm is used to validate whether the optimal solutions of the simulation model resemble the actual rosters as produced by the airline, i.e. the published rosters.

The simulation model is validated using the two problem sets of a small size long-haul aircraft division, each containing 22 different historical rosters. The cost function used for validation, i.e. the '*Validation Function*', only considers two cost parameters of the complete objective function: flexible travel leave (FTL) cost and uncovered pairing cost.

Figures 21 and 22 show the compared results of the rosters as produced using the LP-complete algorithm and the rosters as published by the airline. While the results show that the assignment cost of the rosters produced by the simulation model deviate from those as produced by the airline, the shapes of the two curves are similar. This demonstrates that while the assumptions made in the rostering problem affect the cost of the solution, the behaviour of the simulation model to changing roster environments is comparable to the commercial software program used at the collaborating airline.

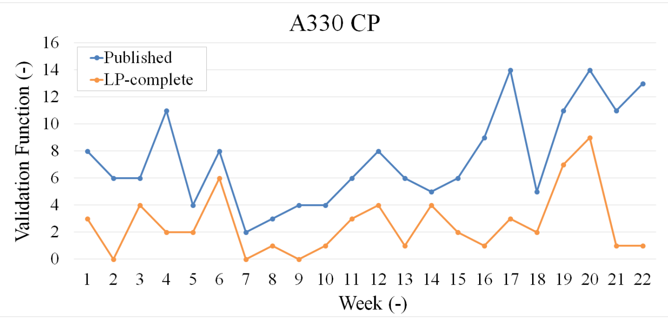

Fig. 21. Model Validation (1)

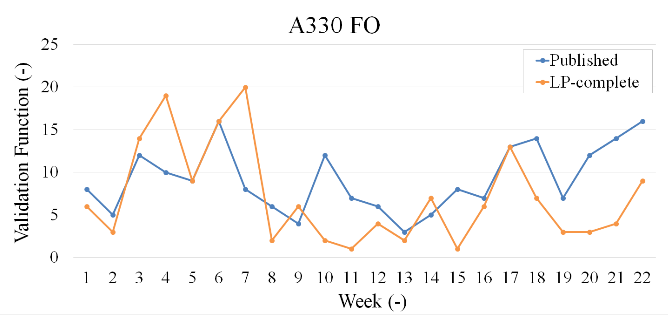

Fig. 22. Model Validation (2)

APPENDIX C: PERFORMANCE ANALYSIS

This appendix provides a more detailed overview of the performance analysis performed on the various developed algorithms.

Solution Quality vs. Computational Time: Figure 23 shows the solution quality and computational time of various configurations of the simulation model compared to two benchmarks: the Hungarian assignment algorithm and an LP algorithm.

Figures 24 and 25 include a couple of other pairing assignment algorithms that have been assessed during this research. Appendix VI provides a description of the considered algorithms. Figure 25 plots the performance of the different algorithms on logarithmic axes in order to include all considered pairing assignment algorithms.

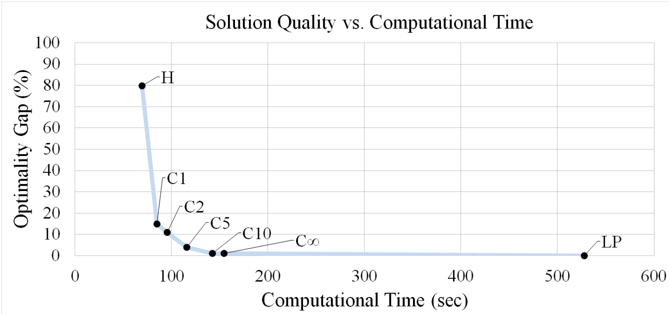

Fig. 23. Performance of simulation model configurations

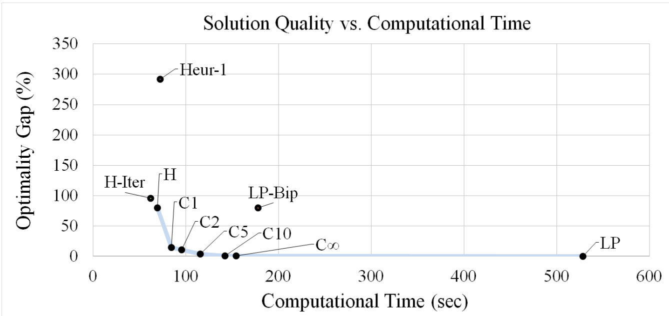

Fig. 24. Performance of various pairing assignment algorithms

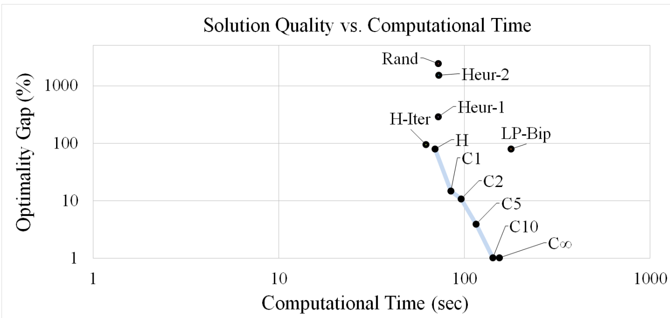

Fig. 25. Performance of all considered assignment algorithms

Computational Time vs. Problem Size: Figure 26 shows the computational time growth for problems of increasing size. In Tabel IV an estimation is made for the order of growth of the different simulation model configurations and benchmarks. Also, the growth function and the R-squared value of the trend lines are presented. Afterwards, several figures present the computational time growth for the various sub-components of the developed simulation model.

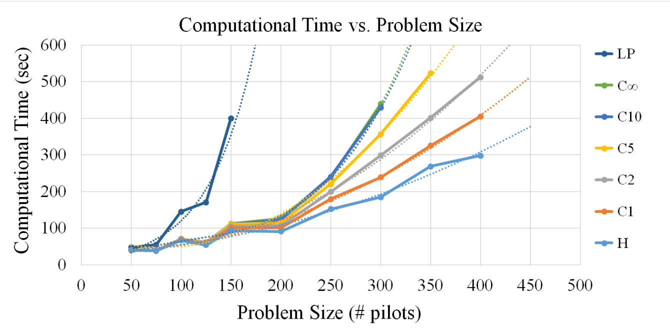

Fig. 26. Computational time growth per simulation model configuration

TABLE IV
COMPUTATIONAL TIME GROWTH OF SIMULATION MODEL CONFIGURATIONS

	Order of Growth	Growth Function	R^2
H	$O(n^2)$	$0,0015x^2 + 0,0865x + 30,732$	0,9805
C1	$O(n^2)$	$0,0026x^2 - 0,1039x + 40,224$	0,9916
C2	$O(n^2)$	$0,0038x^2 - 0,3353x + 50,265$	0,9929
C5	$O(n^2)$	$0,0064x^2 - 1,0279x + 89,226$	0,9888
C10	$O(n^3)$	$0.00004x^3 - 0,0142x^2 + 1,9979x - 35,522$	0,9909
C∞	$O(n^3)$	$0,00005x^3 - 0,0158x^2 + 2,221x - 44,121$	0,9912
LP	$O(e^n)$	$14,024e^{0,0216x}$	0,9453

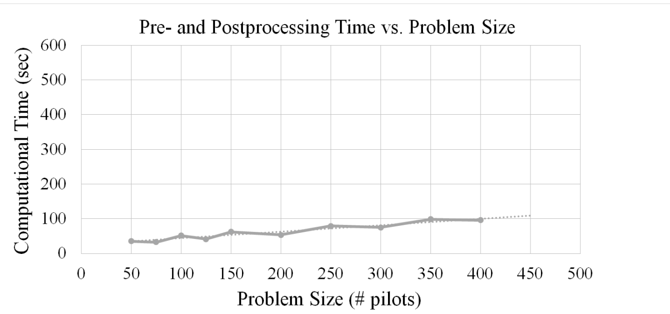

Fig. 27. Pre- and postprocessing time growth over problem size

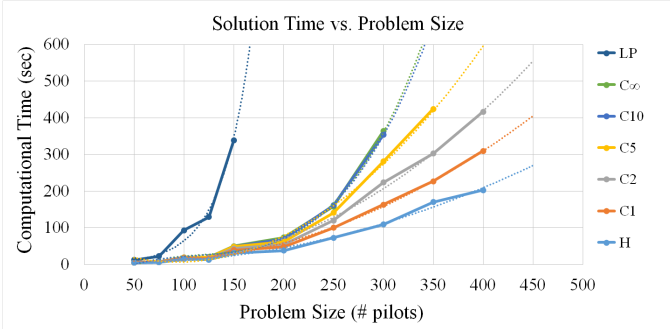

Fig. 28. Solution time growth over problem size

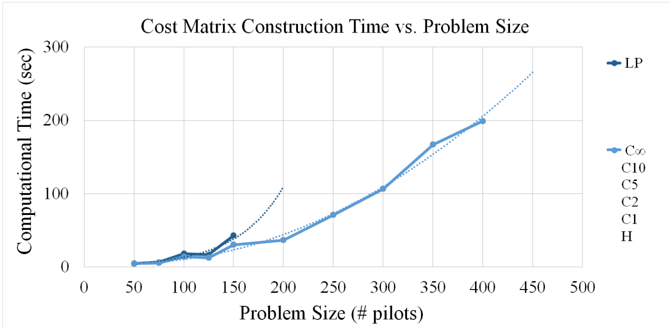

Fig. 29. Cost matrix construction time growth over problem size

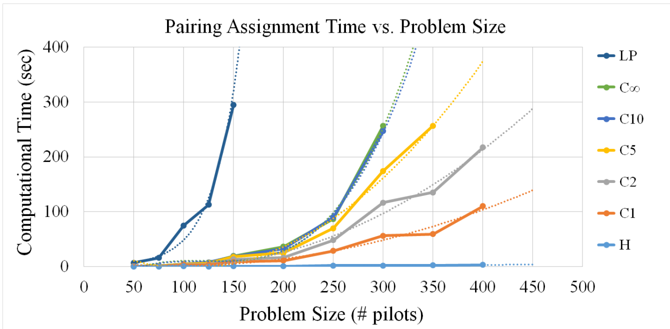

Fig. 30. Pairing assignment time growth over problem size

APPENDIX D: FTL SCENARIO ANALYSIS INTERPRETATION

This appendix presents how the results of the scenario analysis on flexible travel leave (FTL) can be interpreted in order to define the benefit of having FTL pilots, i.e. pilots with the possibility to remove the last day of their travel leave. Below, the results of the FTL scenario analysis can be found.

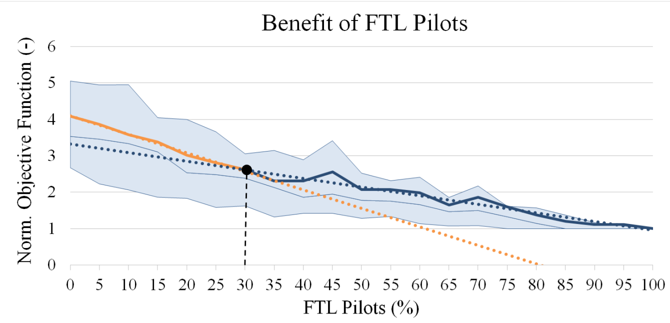

Fig. 31. The impact of having FTL pilots on roster efficiency

From the results of the scenario analysis, the following insights have been obtained:

TABLE V
RESULTS OF THE FTL SCENARIO ANALYSIS EXPERIMENT

No. of Pilots	140
Average No. of Days of Inefficiency @ 100% FTL Pilots	8,1
Linear Trend 0-30%	y = -0,2544x + 4,357
R of Trend 0-30%	0,995
Linear Trend 30-100%	y = -0,1182x + 3,445
R of Trend 30-100%	0,9443

From the results, an estimation has been made of the actual benefit in terms of cost of having FTL pilots. In this calculation, the average pilot salary cost has been estimated to be around €200.000 and a pilot is expected to have 5 working days per week on average over a year. Using these figures, below a step-by-step calculation of the benefit per extra FTL pilot is calculated for the two ranges, 0-30% and 30-100%.

TABLE VI
EXAMPLE CALCULATION OF THE BENEFIT OF FTL PILOTS

	0-30% FTL Pilots	30-100% FTL Pilots
Delta Norm. Obj F per Delta 5% of FTL Pilots	-0.25	-0.12
Average No. of Days of Inefficiency @100% FTL Pilots	8,1	8,1
Delta No. of Days of Inefficiency per Delta 5% of FTL Pilots	2,03	0,96
Number of Pilots	140	140
Delta No. of Days of Inefficiency per 7 extra FTL Pilots	2,03	0,96
Delta No. of Days of Inefficiency per extra FTL Pilot (per week)	0,29	0,14
Delta No. of Days of Inefficiency per extra FTL Pilot (per year)	15,1	7,3
Salary cost of Pilot (per working day)	€767	€767
Benefit of extra FTL Pilot (per year)	€11.500	€5.600